Self-Supervised Learning in Vision

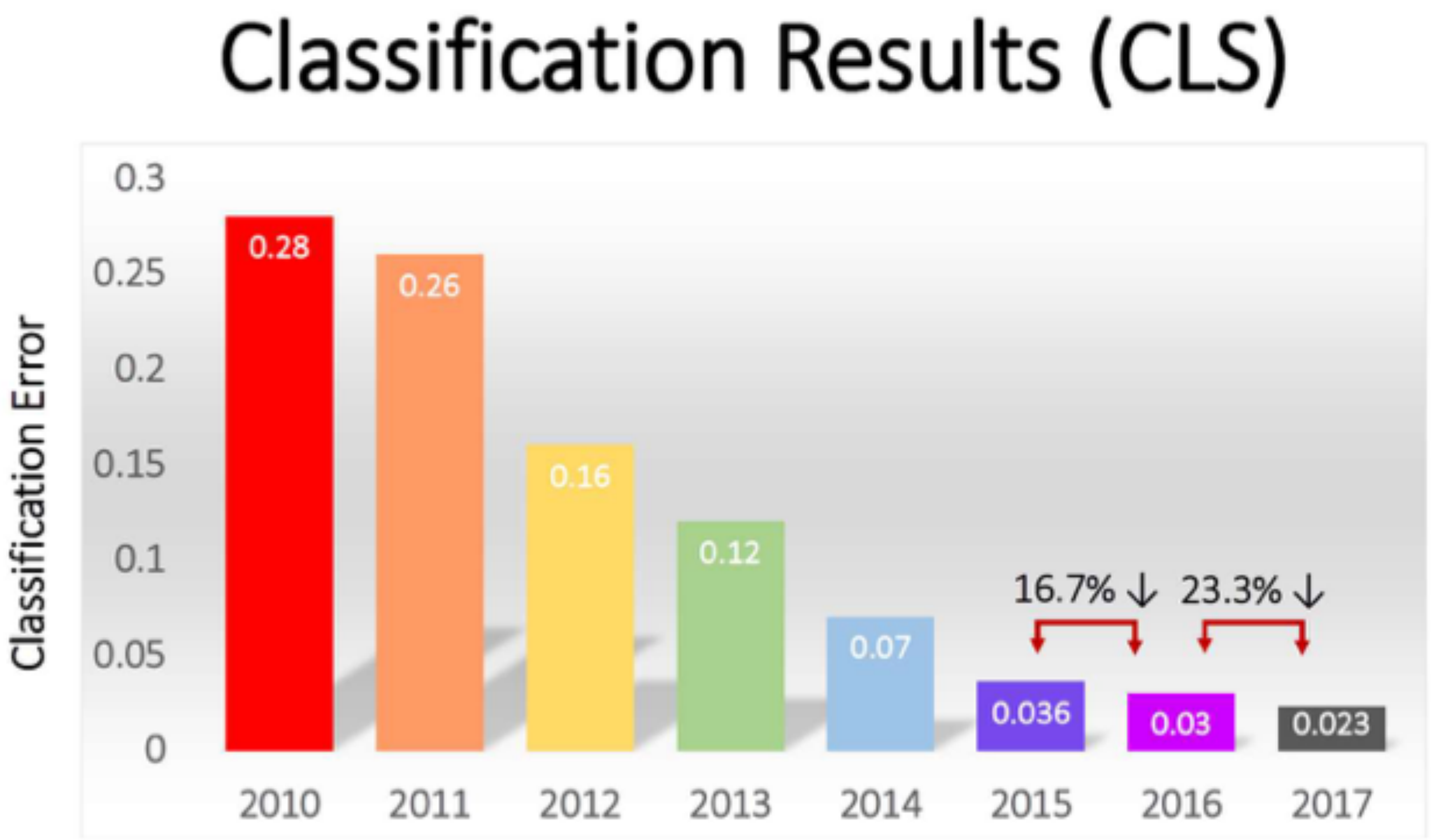
Lecture 6

Rob Fergus

Slides from Ishan Misra, Naiyan Wang & many others.

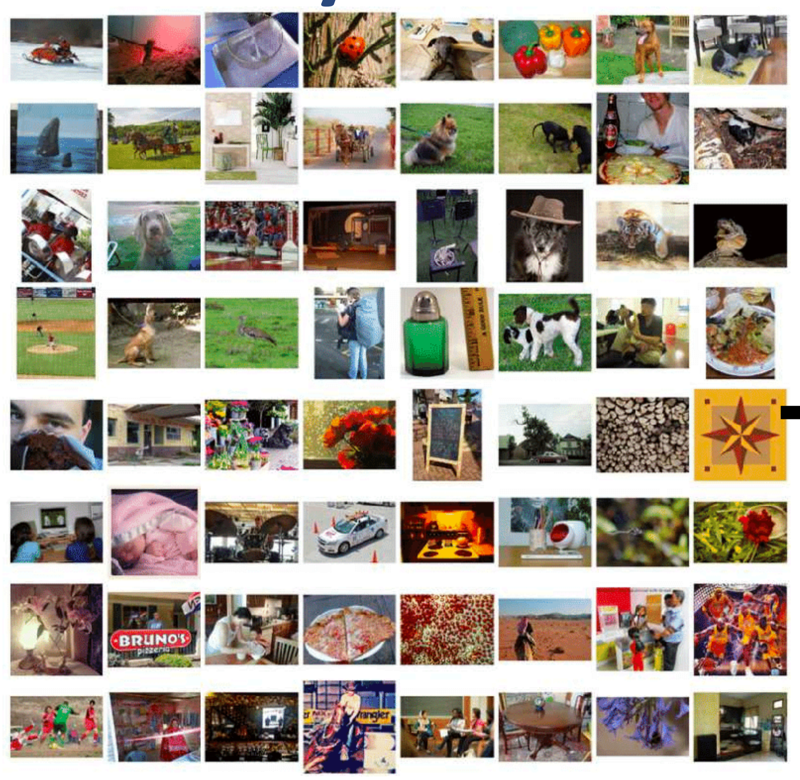
Success story of supervision

ImageNet Challenge

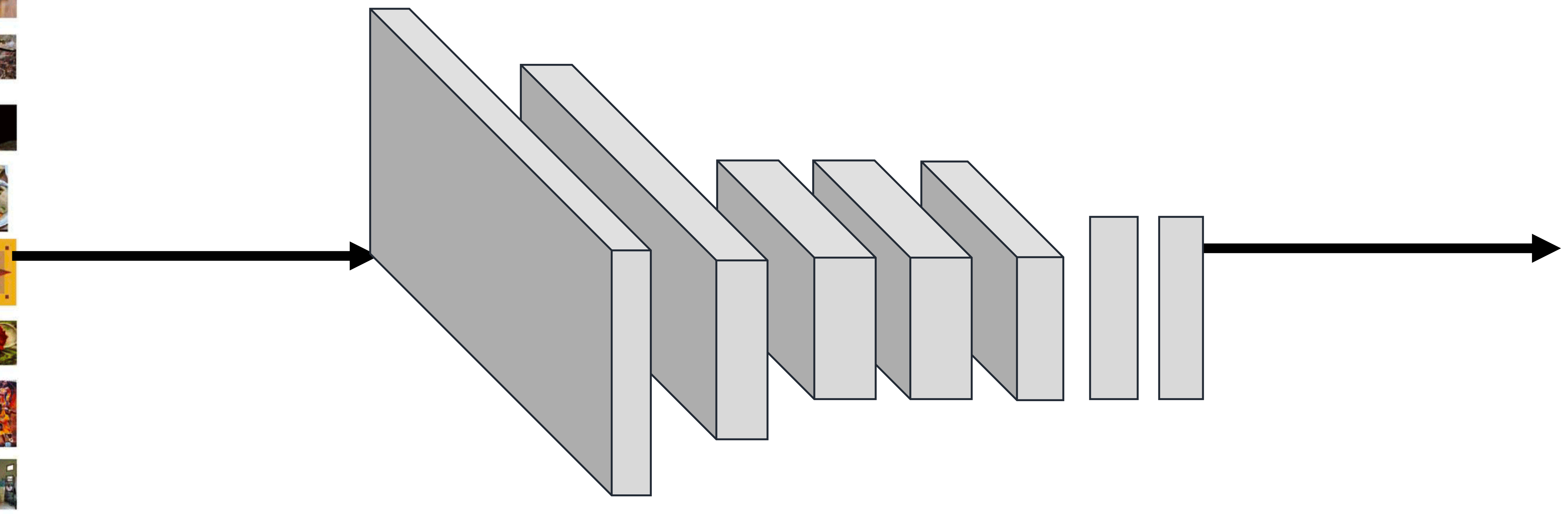


Success story of supervision: Pre-training

- Features from networks pre-trained on ImageNet can be used for a variety of different downstream tasks



Images from ImageNet (Pre-train)



husky, terrier, tench, ...

ConvNet

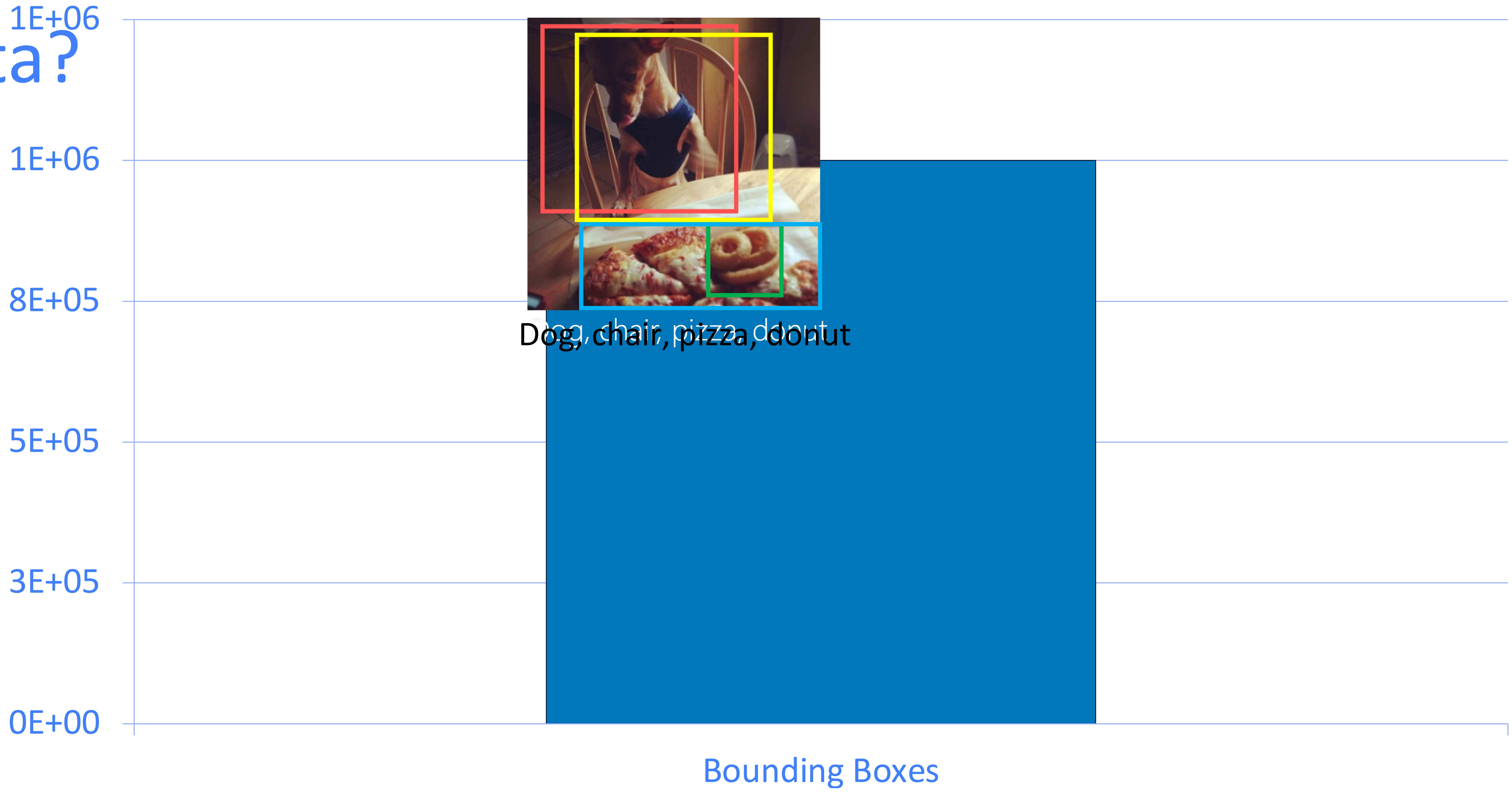
Learn a representation

Success story of supervision: Recipe for good solutions

- Pre-train on a large supervised dataset.
- Collect a dataset of "supervised" images
- Train a ConvNet

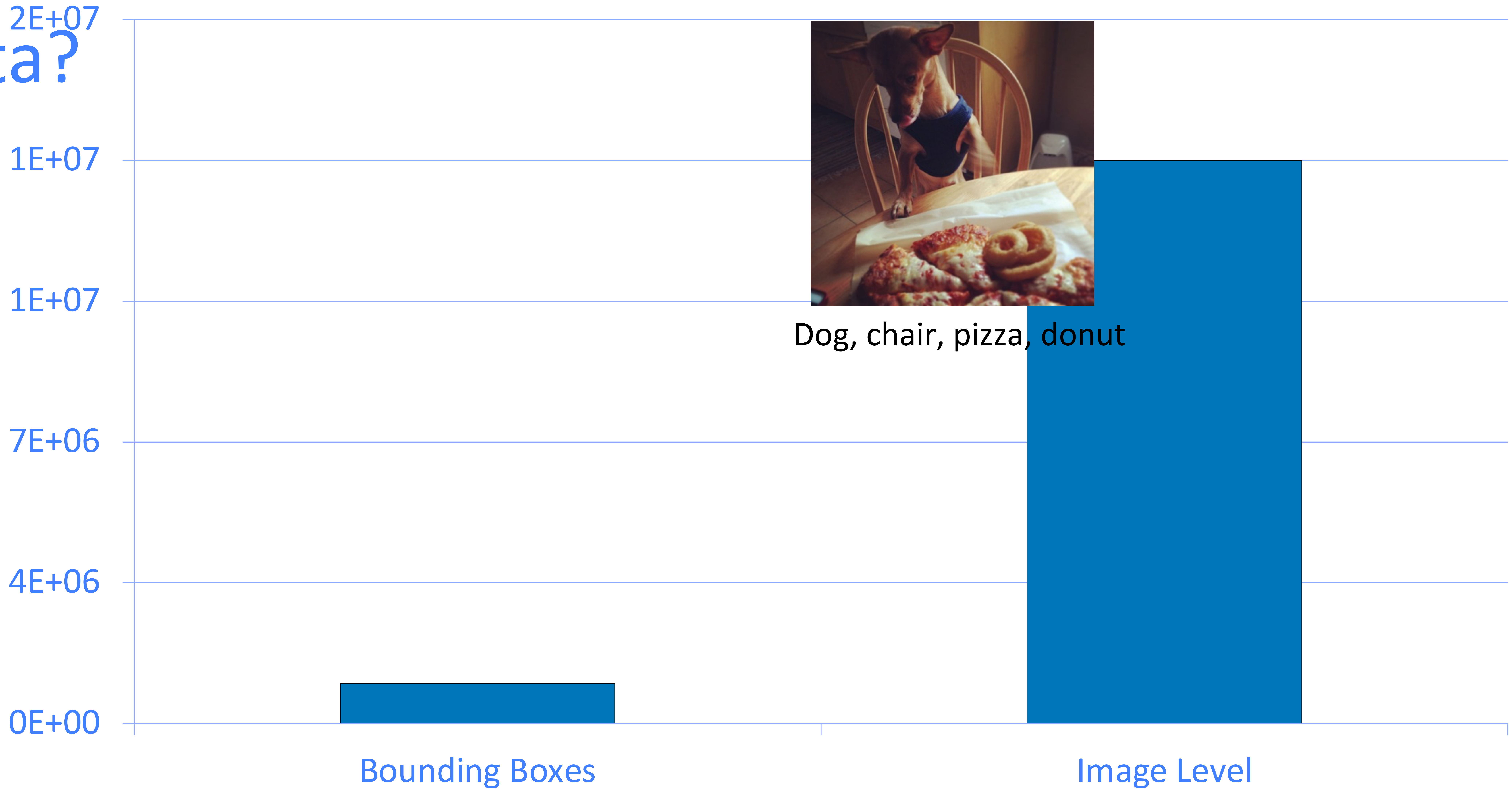
Can we get labels for all data?

Can we get labels for all data?



Stats from Pawan Kumar at Oxford

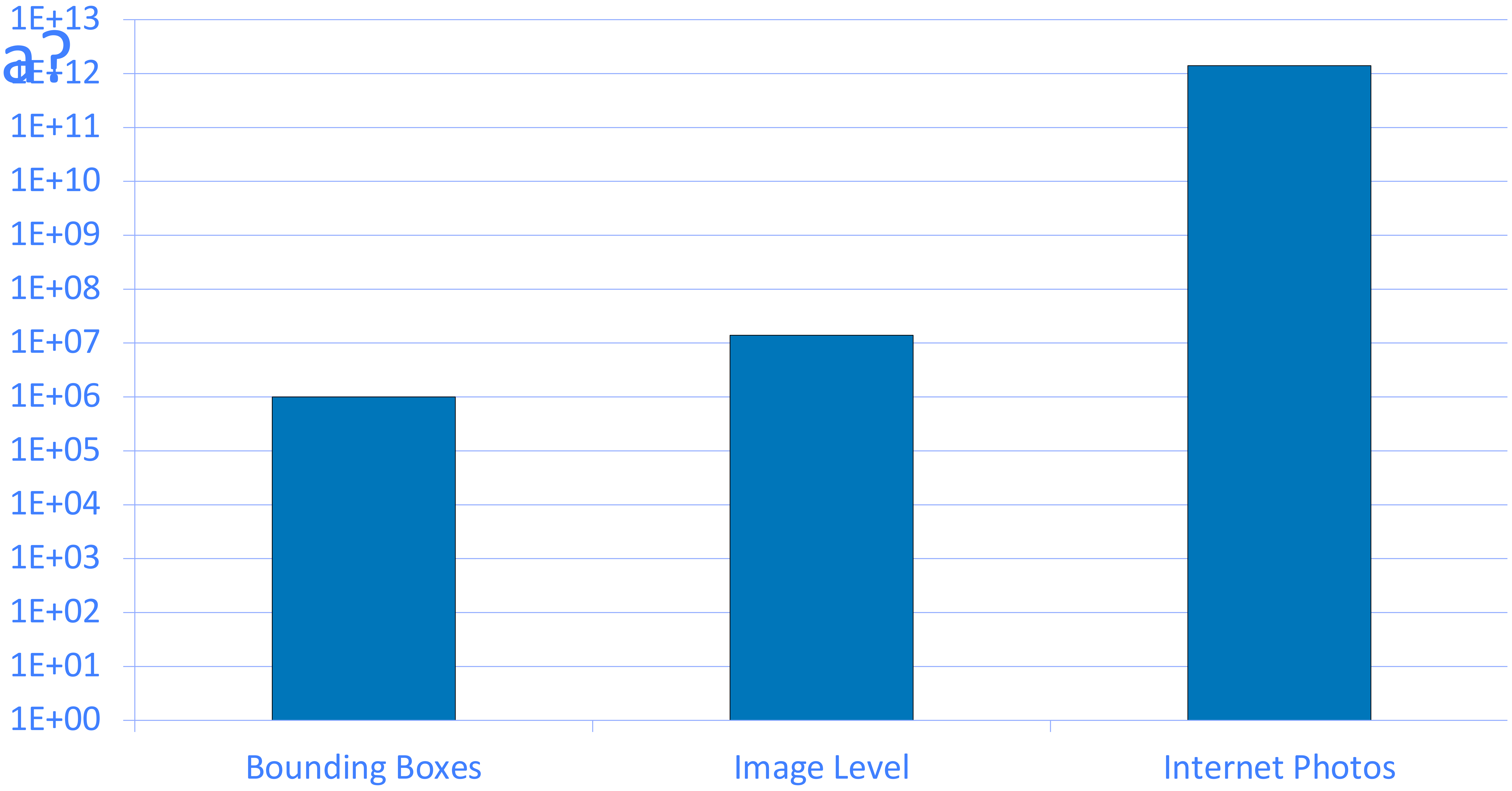
Can we get labels for all data?



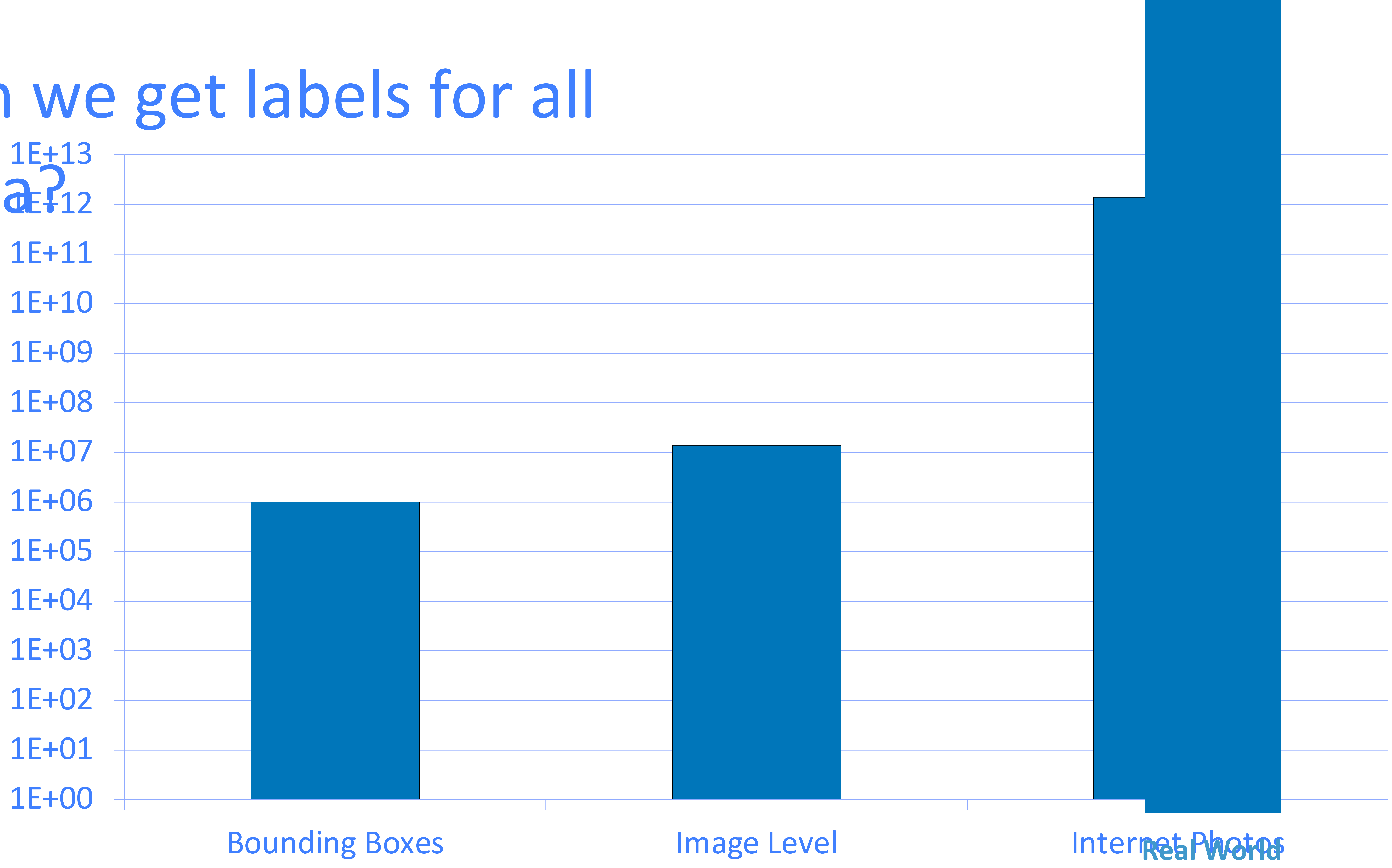
Can we get labels for all data?

[forbes.com](https://www.forbes.com/sites/bernardmarr/2018/05/21/how-much-data-do-we-create-every-day-the-mind-blowing-stats-everyone-should-read/)
https://www.forbes.com/sites/bernardmarr/2018/05/21/how-much-data-do-we-create-every-day-the-mind-blowing-stats-everyone-should-read/

Can we get labels for all data?



Can we get labels for all data?

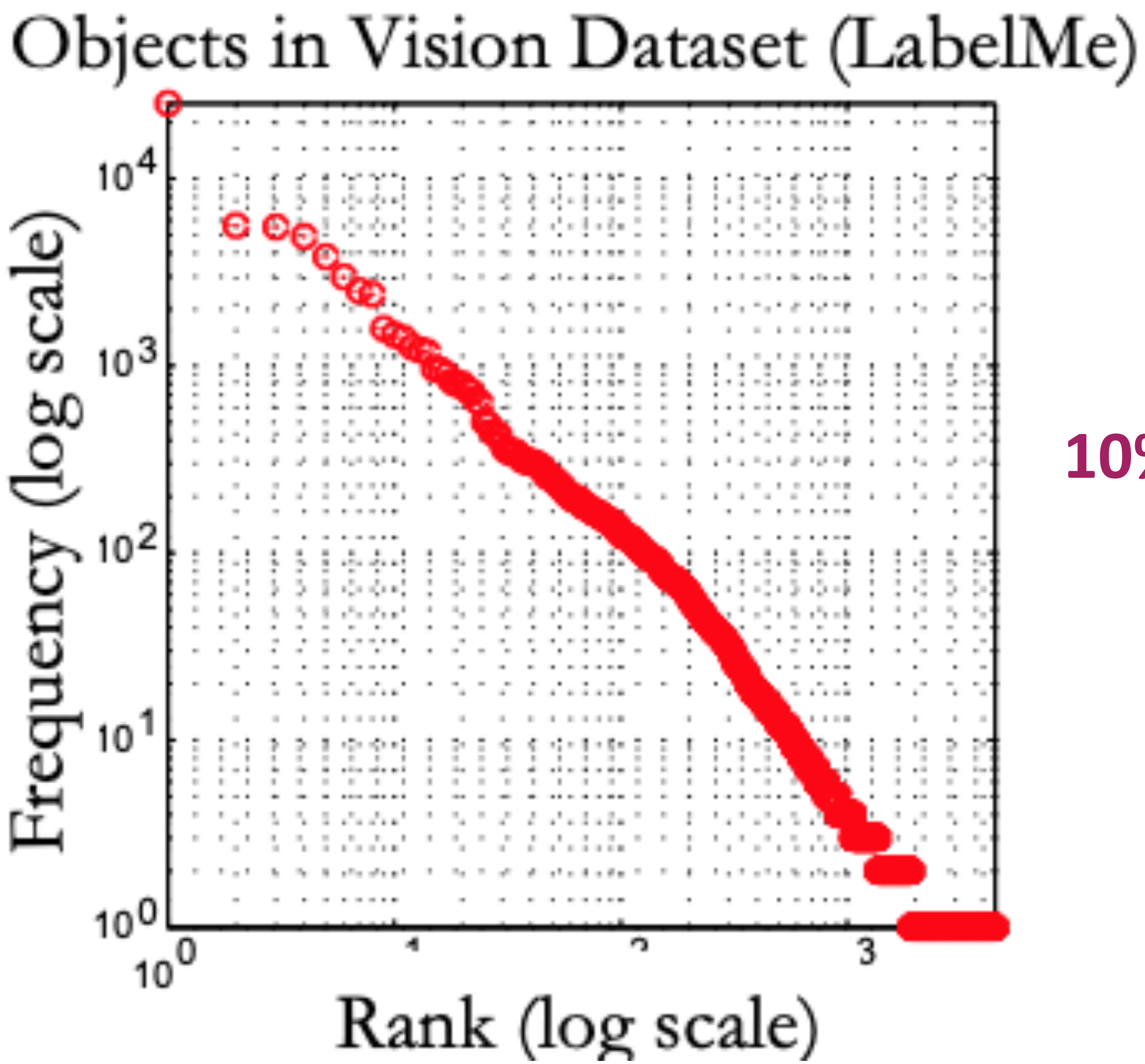


ImageNet (14 million images) needed 22 human years to label

Can we get labels for all data?

- What about complex concepts?
  - Video?
- Labelling cannot scale to the size of the data we generate

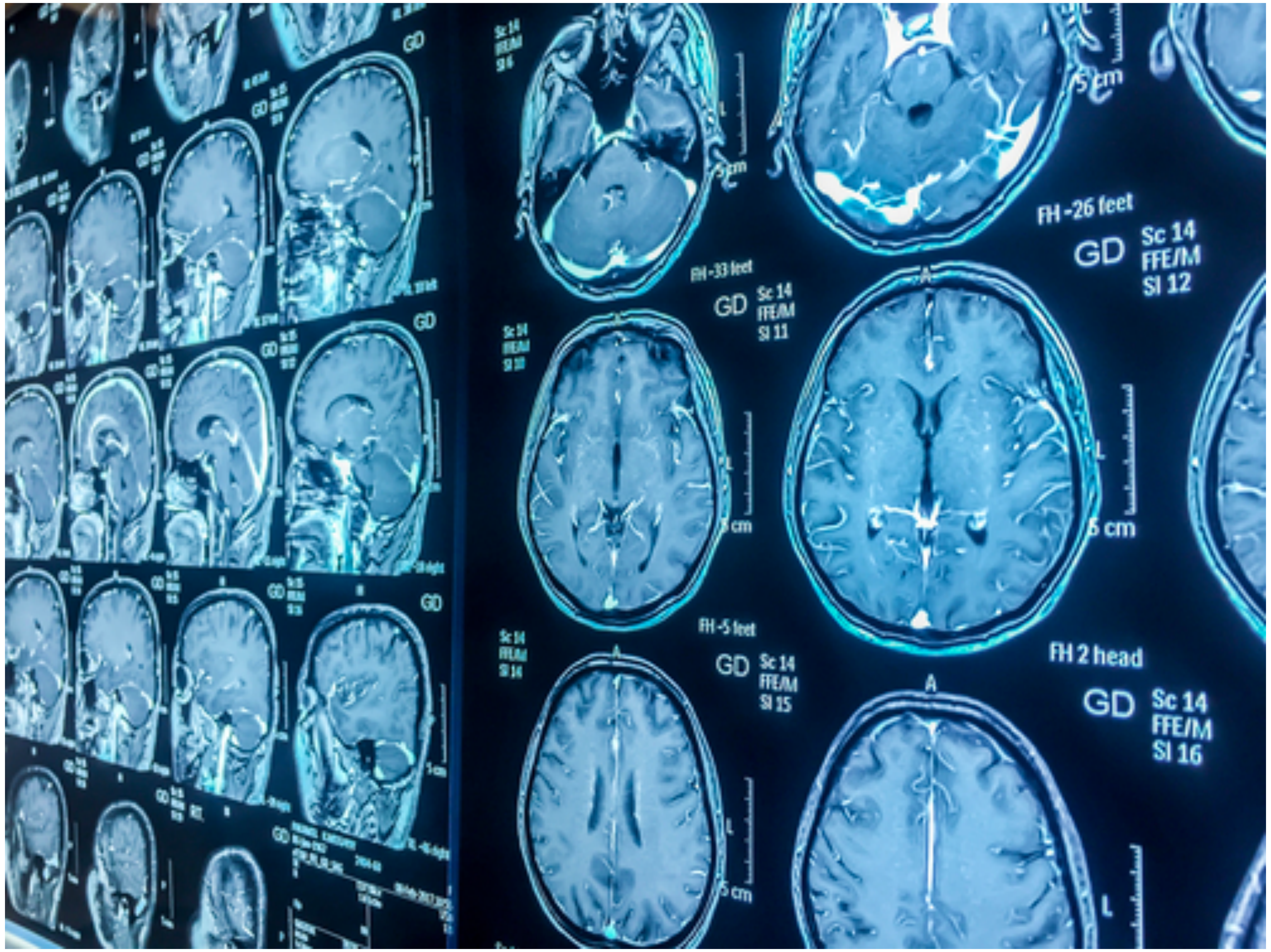
Rare concepts?



10% of the classes account for 93% of the data

Slide credit: Rob Fergus

Different Domains?



ImageNet pre-training may not work

Arguments for Unsupervised Learning

- Want to be able to exploit unlabeled data
  - Vast amount of it often available
  - Essentially free

- Good regularizer for supervised learning
  - Helps generalization
  - Transfer learning
  - Zero/one/few - shot learning

Unsupervised Learning

- Biological argument [from G. Hinton]:
  - Our brains have 10^15 connections
  - We live for 10^9 secs
  - Need 10^6 bits/sec
  - Insufficient information from occasional high level label
  - Only source with enough information is input itself

- Challenging problem: big focus on many DL groups

Historical Note

- Deep Learning revival started in ~2006
  - Hinton & Salakhudinov Science paper on RBMs
- **Unsupervised** Deep Learning was focus from 2006-2012
- In ~2012 great results in vision, speech with **supervised** methods appeared
  - Initially less interest in unsupervised learning
  - By focus once more on unsupervised learning

Overview of Unsupervised Perspectives

- Given just data {X}
  - Unlike supervised learning there are no **provided** labels {Y}

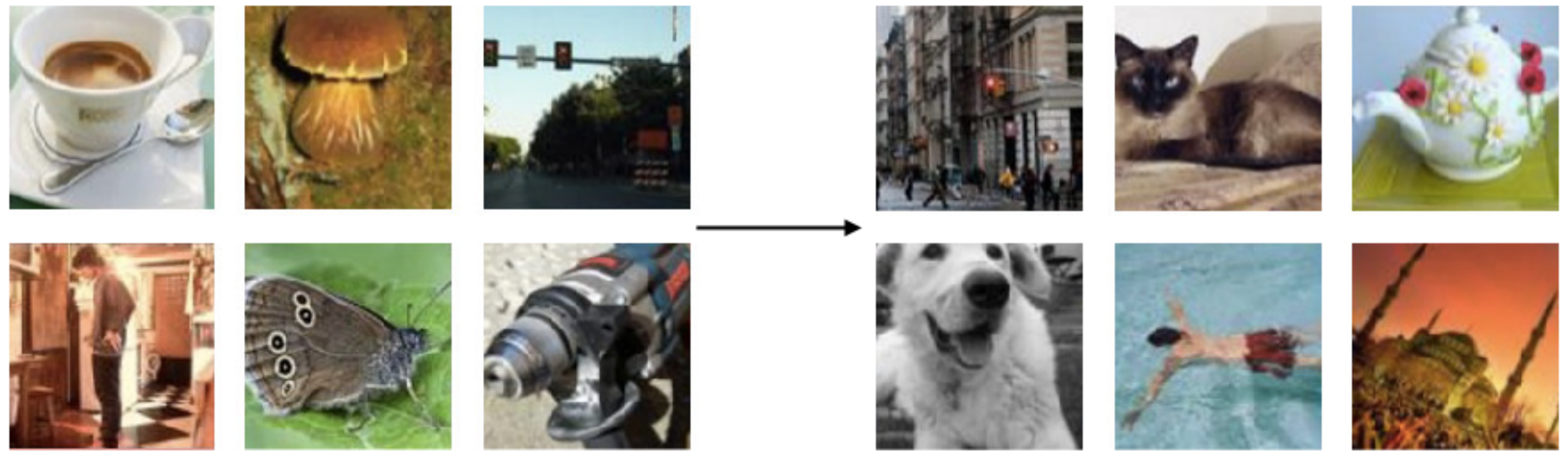
1. Density modeling, i.e. build model of p(X)
   - Enables sampling of new data
   - Evaluate probability of a data point
   - Can be conditional model, e.g. $p(X_t \mid X_{t-1},\ldots)$
   - Requires (deep) generative architectures [HARD]
   - Generating pixels not necessarily optimal for learning representations for downstream tasks

1. Density Modeling

- Have access to $x \sim p_{data}(x)$ through training set
- Want to learn a model $x \sim p_{model}(x)$
- Want p_{model} to be similar to p_{data}:

Samples from true data distribution have high likelihood under p_{model}

Samples drawn from p_{model} reflect structure of p_{data}



Training examples Model samples

2. “Self supervised” learning

- Also unsupervised but…
- Find supervision signal y **within the input data**
- This signal is then used as a target in *discriminative model*:

$$\begin{aligned} y : \mathcal{X} &\to \mathcal{Y} \\ x &\mapsto y(x) \end{aligned}$$

- Allows the use of <u>standard supervised learning losses and architectures</u>

$$\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} \ell(f_{\theta}(x_i), y(x_i))$$

- Pre-training of representation for subsequent task
- Typically involves some insight into domain to pick y

Class Project

- **Everyone must come and discuss proposed ideas with me at some point during office hours.**
- Simple option is to reimplement existing approach/paper.
  - Cannot just copy existing repo, but can use it for debugging.
- Alternatively, you can extend existing approach/paper.
  - Can use existing repo, but extension must be significant
- More ambitious (more marks): Implement new idea from scratch
  - Can use existing models/code but must have significant novelty
- Project abstracts due Oct 15th (email to me).
  - Email paragraph summarizing your project plan to fergus@cs.nyu.edu.
  - Please be sure to include the names of people on your project.

Class Project details

Team of 2-3 people [1 person not allowed; 3 is hard max]

PyTorch preferred

Project video:

- 2 minute clip explaining your project
- Voice over slides

Project Report

- Format: 4-8 page conference paper style report on your project (please don't waffle)
- Intro (with refs to related work) [~1 page]
- Method (be sure to cite any code/pre-trained models) [~2 pages]
- Experiments (**must have plots/results figures**; also should have baselines; ideally some kind of ablation experiments too) [~2-4 pages]
- Discuss (brief) [~0.5 pages
- See examples: http://openaccess.thecvf.com/CVPR2018.py
- Zip of source code or link to Github (please ensure you give access to robfergus)

Class Project details (2)

- Deadline is Thursday Dec 17th midnight [Hard deadline]
  - Feel free to turn in earlier
  - Will try to grade them and compute final grades by Christmas

- Grading (49% of total grade for class)
  - Novelty / Technical difficulty of problem [15%]
  - Quality of Results [15%]
  - Quality of implementation [5%]
  - Quality of writeup & video presentation [14%]
  - How many people in your group

Class Project General Advice

- Please make sure you have *something* working, even if you don’t achieve overall goal
- Even a small part of an ambitious project can be OK
- So **please have a safe plan B option** in mind
- Expect all projects to train something, i.e. must use b-prop at some point
- Just evaluating existing models is NOT OK.
- Cluster gets busy at end of semester -- please don’t leave it all to last moment.

Self-supervised learning in computer vision

Ishan Misra

facebook
Artificial Intelligence Research

With slides from Andrew Zisserman, Carl Doersch

What is "self" supervision?

- Obtain labels from the data itself by using a "semi-automatic" process

What is "self" supervision?

- Obtain labels from the data itself by using a "semi-automatic" process

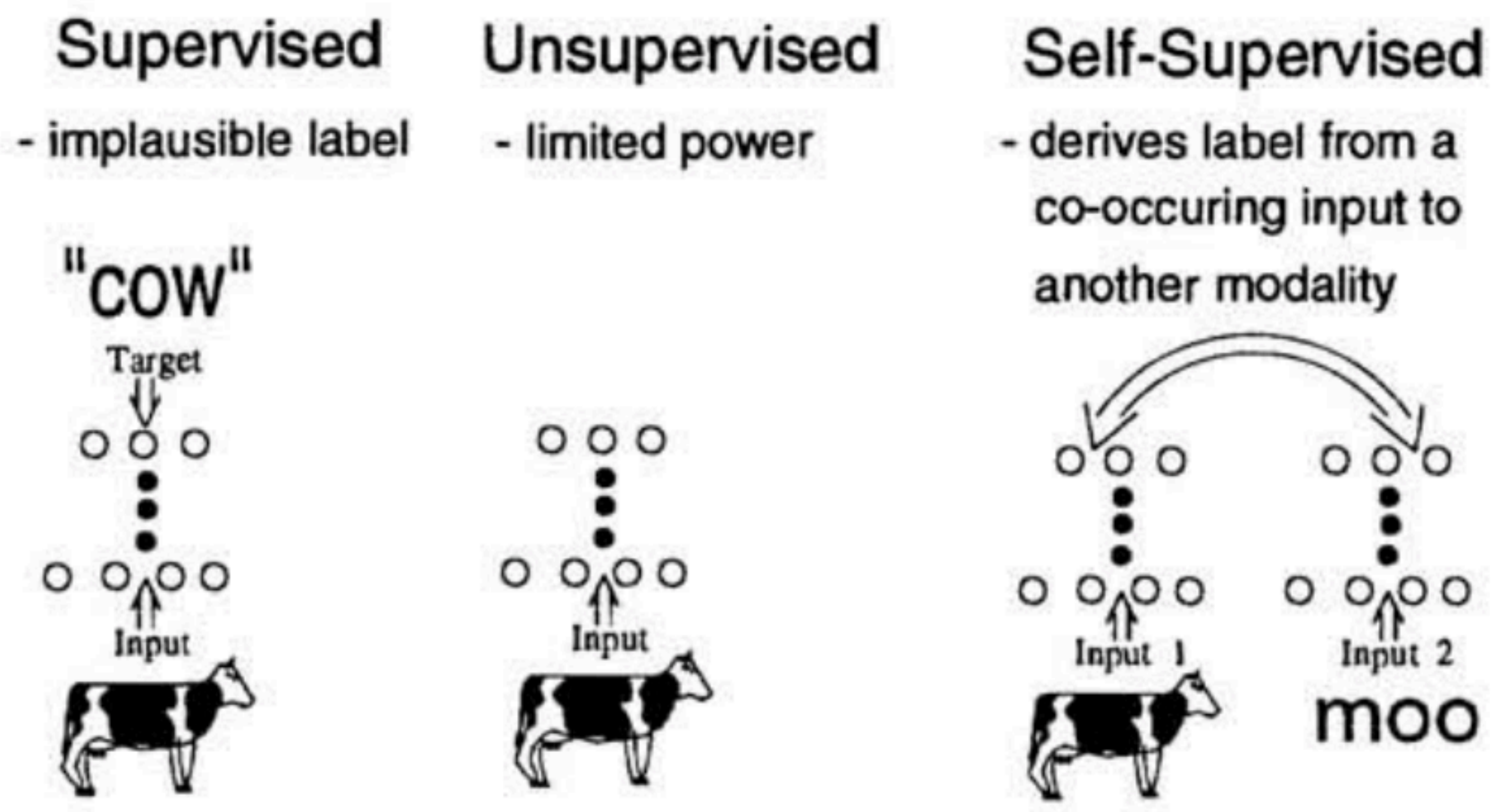
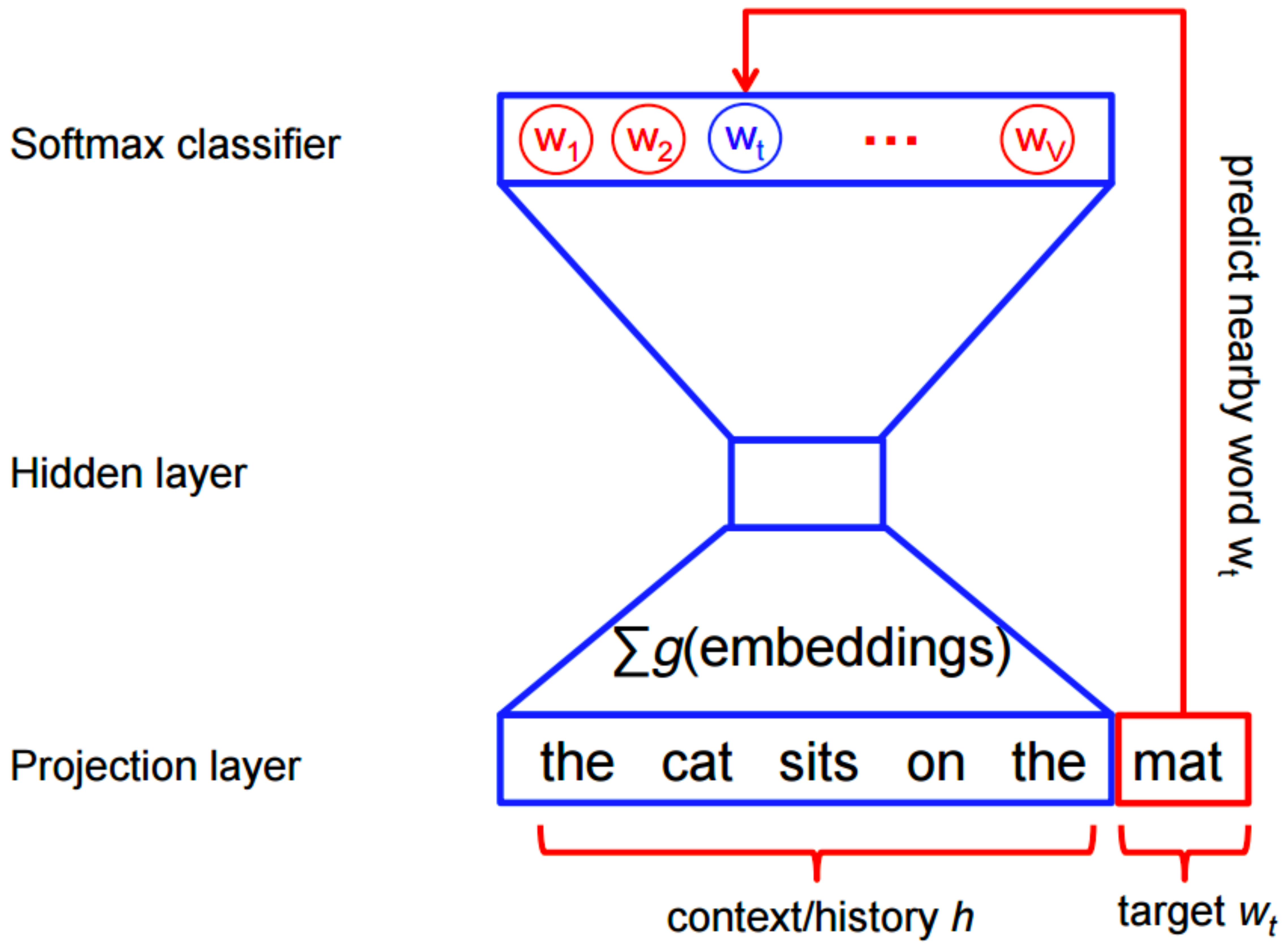


Image: Learning classification with Unlabeled Data - de Sa, 1994

Word2vec

- Fill in the blanks



word2vec - Mikolov et al.
Image by Julian Gilyadov

Success of self-supervised learning in NLP

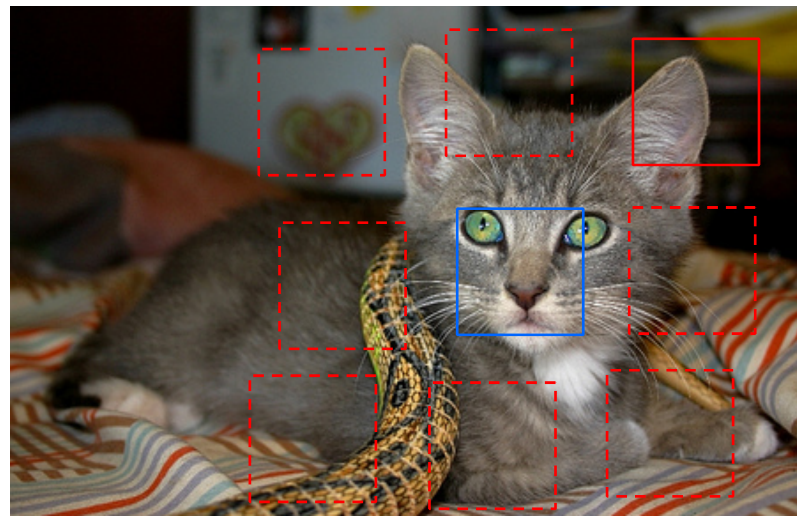
- Fill in the blanks is a powerful signal to learn representations

- Sentence/Word representations: BERT - Devlin et al., 2018

Why self supervision?

- Helps us learn using observations and interactions
- Does not require exhaustive annotation of concepts
- Leverage multiple modalities or structure in the domain

Range of self-supervised systems

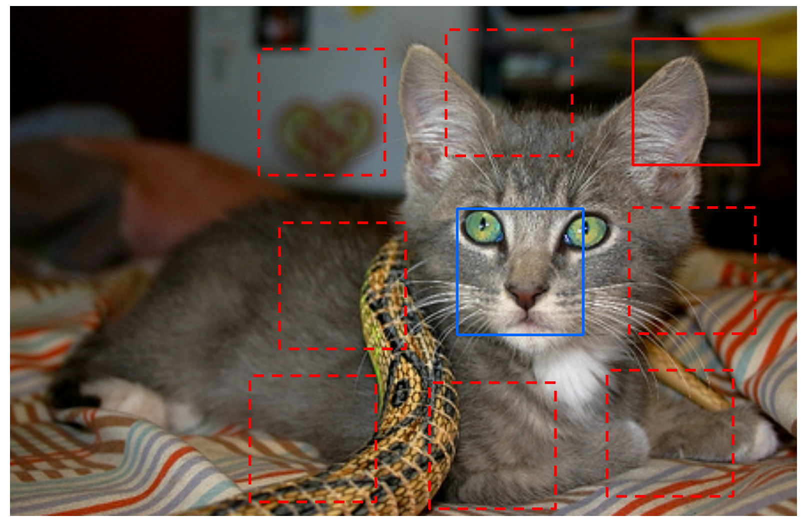


images

richer data

- Doersch et al. (2015)
- Zhang et al. (2016)
- Zhang et al. (2017)
- Noroozi et al. (2016)
- Pathak et al. (2016)

Range of self-supervised systems



images

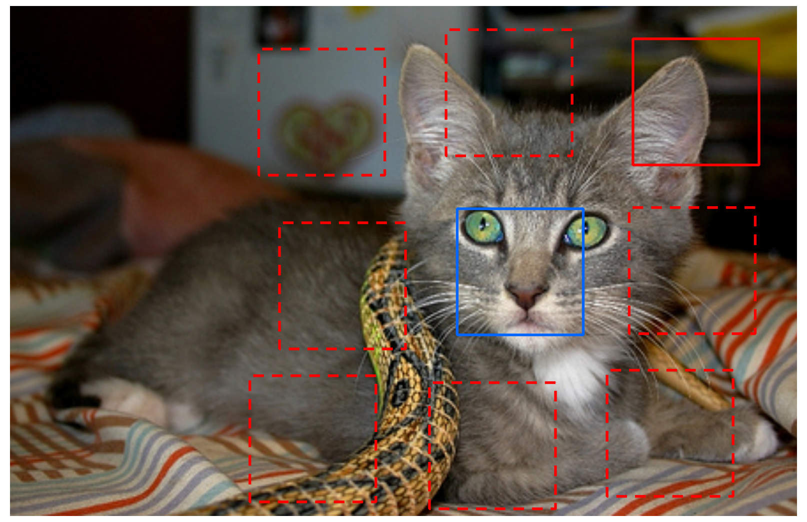
videos

richer data

- Doersch et al. (2015)
- Zhang et al. (2016)
- Zhang et al. (2017)
- Noroozi et al. (2016)
- Pathak et al. (2016)

- Wang et al. (2015)
- Misra et al. (2016)
- Pathak et al. (2017)

Range of self-supervised systems



images videos sound & depth

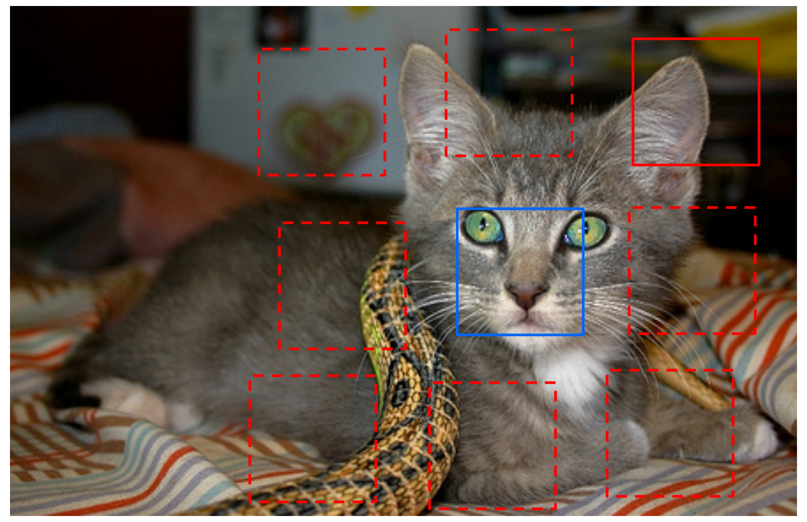
richer data

- Doersch et al. (2015)
- Zhang et al. (2016)
- Zhang et al. (2017)
- Noroozi et al. (2016)
- Pathak et al. (2016)

- Wang et al. (2015)
- Misra et al. (2016)
- Pathak et al. (2017)

- Owens et al. (2016)
- Zhang et al. (2017)
- Bansal et al. (2016)

Range of self-supervised systems



images

videos

sound & depth

actions

richer data

- Doersch et al. (2015)
- Zhang et al. (2016)
- Zhang et al. (2017)
- Norouzi et al. (2016)
- Pathak et al. (2016)

- Wang et al. (2015)
- Misra et al. (2016)
- Pathak et al. (2017)

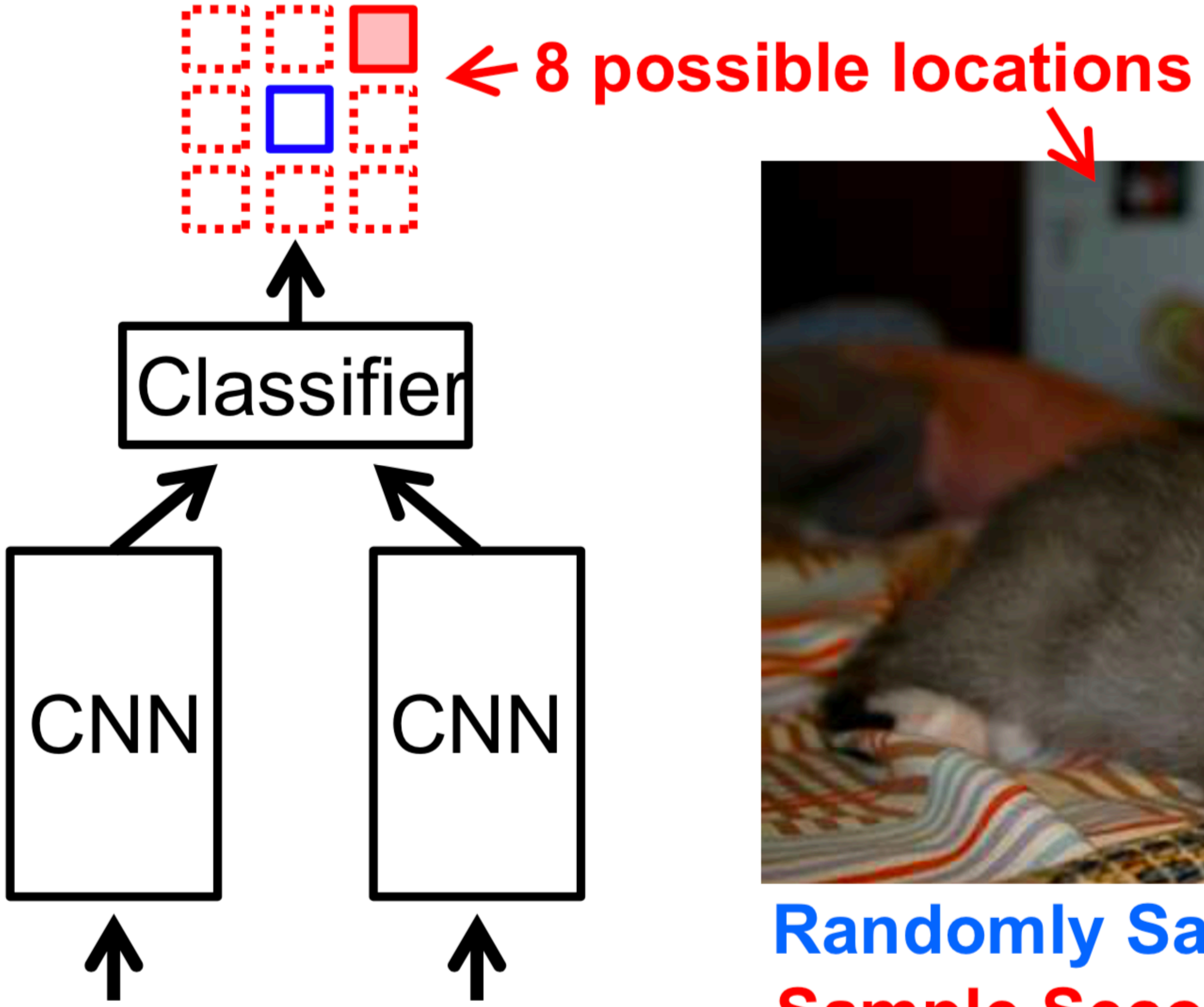
- Owens et al. (2016)
- Zhang et al. (2017)
- Bansal et al. (2016)

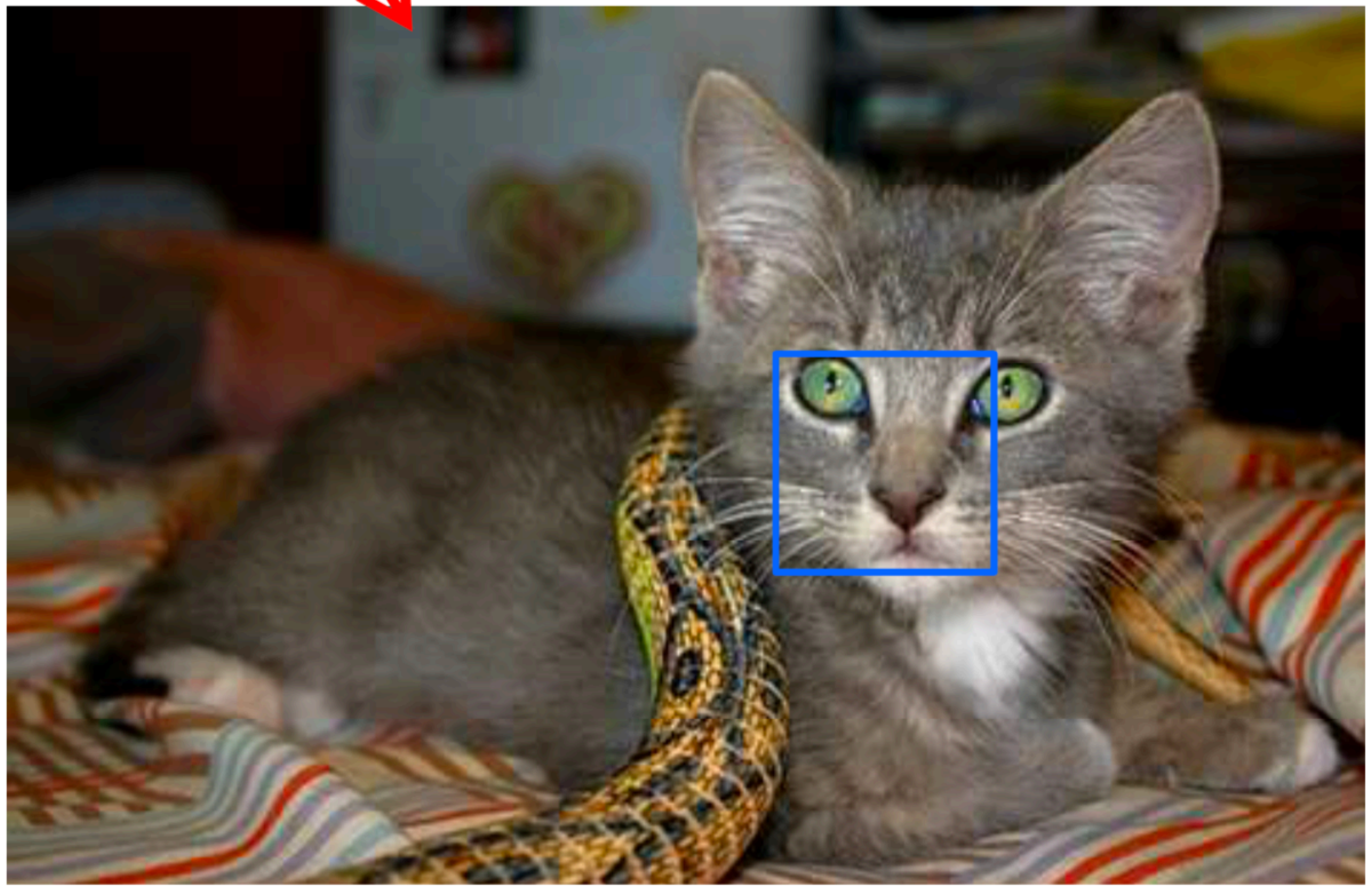
- Agarwal et al. (2015)
- Jayaraman et al. (2015)
- Pinto et al. (2016)
- Agarwal et al. (2016)
- Pinto et al. (2017)
- Pinto et al. (2016)

Self-supervision in computer vision

- Using images
- Using video
- Using video and sound

From Images: Relative position of patches





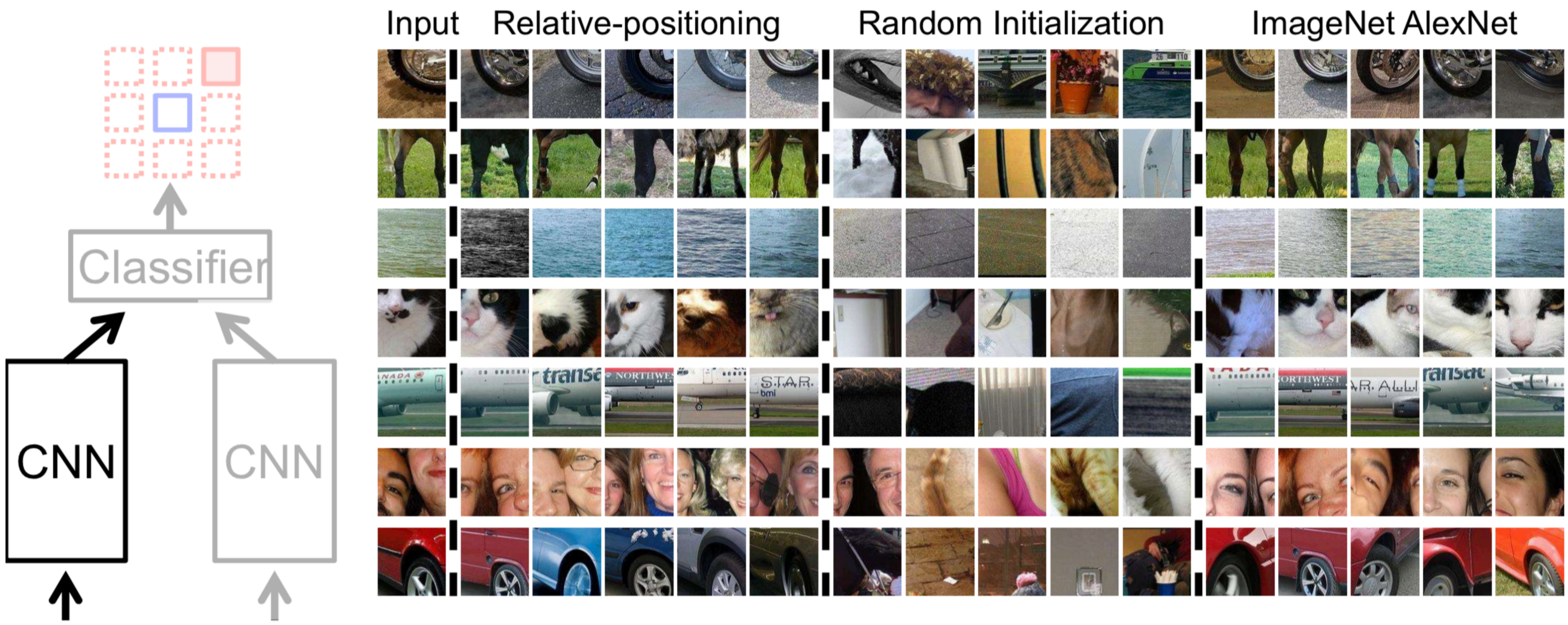
Randomly Sample Patch
Sample Second Patch

Unsupervised visual representation learning by context prediction, Carl Doersch, Abhinav Gupta, Alexei A. Efros, ICCV 2015

From Images: Relative position of patches

Unsupervised visual representation learning by context prediction, Carl Doersch, Abhinav Gupta, Alexei A. Efros, ICCV 2015

Relative Position: Nearest Neighbors in features

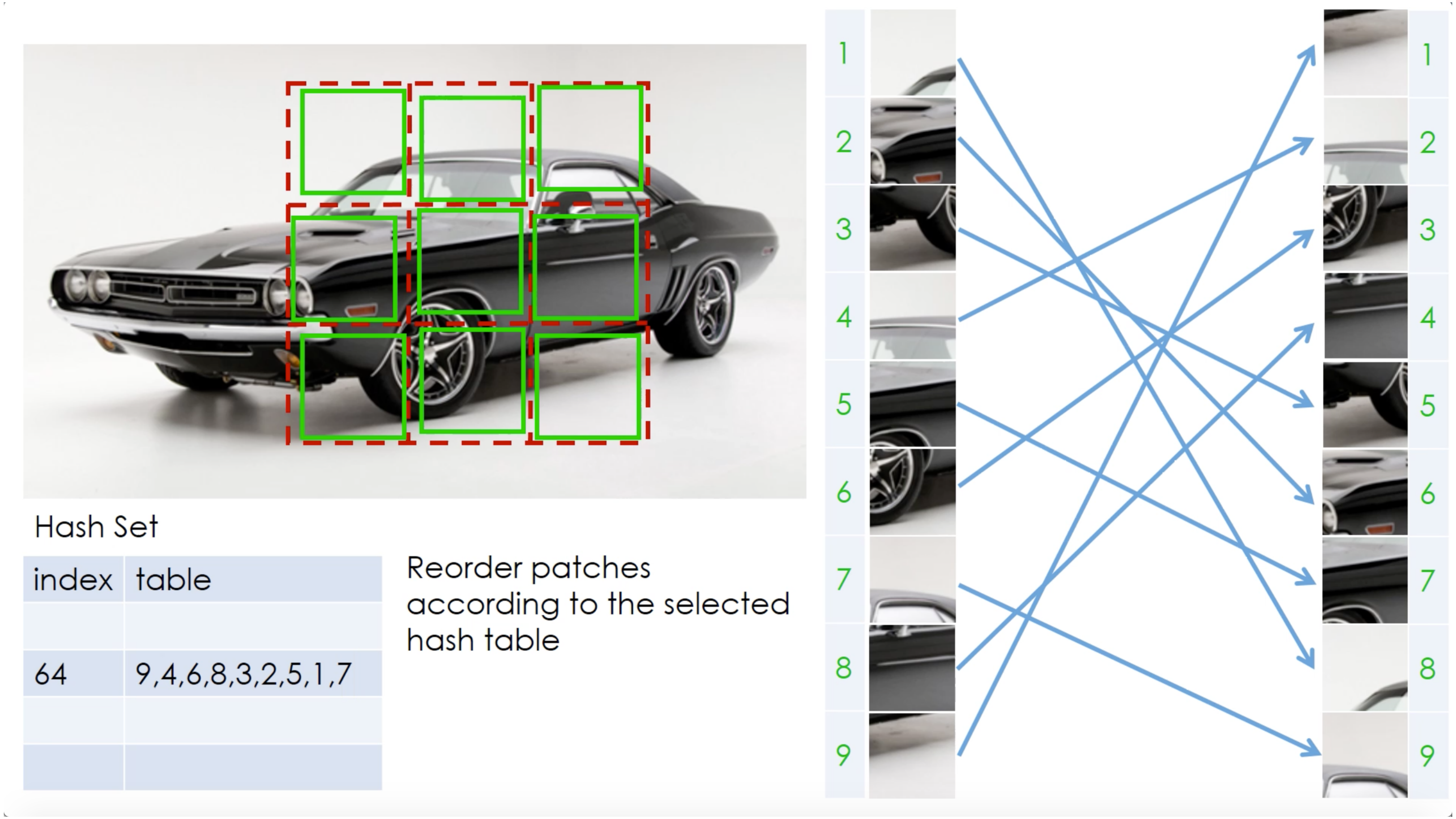


Unsupervised visual representation learning by context prediction, Carl Doersch, Abhinav Gupta, Alexei A. Efros, ICCV 2015

Unsupervised Learning of Visual Representations by Solving Jigsaw Puzzles, Noorozi et al. (2016)

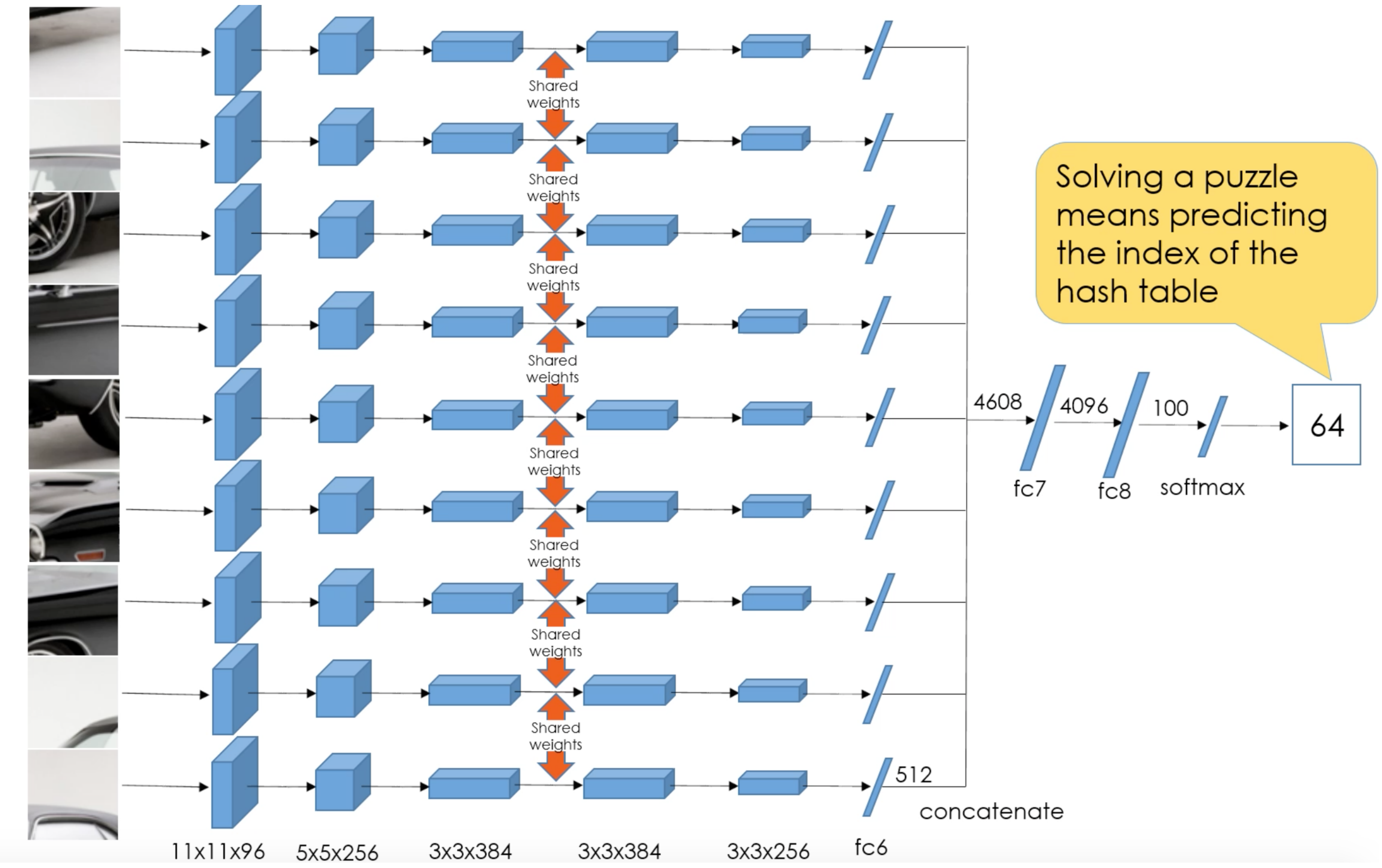
What do we learn when we solve a Jigsaw puzzle?

[Noorozi et al. (2016)]



index	table
64	9,4,6,8,3,2,5,1,7

[Noorozi et al. (2016)]



[Noorozi et al. (2016)]

- Visualization of filters

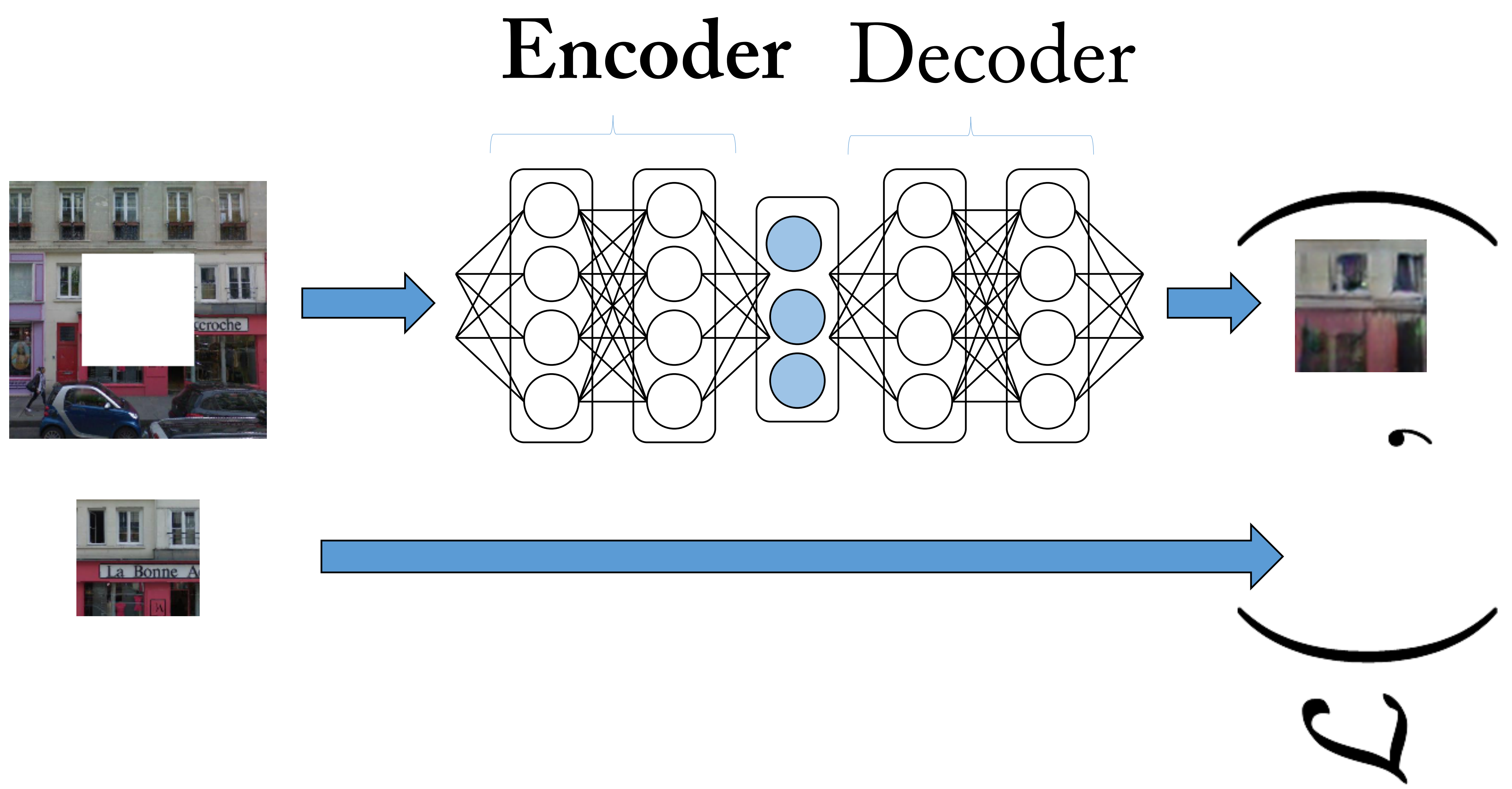
Noroozi, M., & Favaro, P. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV 2016*.

Feature Learning by Inpainting

[Context Encoders: Feature Learning by Inpainting, Pathak et al. (2016)]

[Pathak et al. (2016)]

Context Encoders



- Encoder can be substituted with any network architecture like AlexNet etc.
- Decoder is a set of UpConv/deconv/frac-strided-conv layers

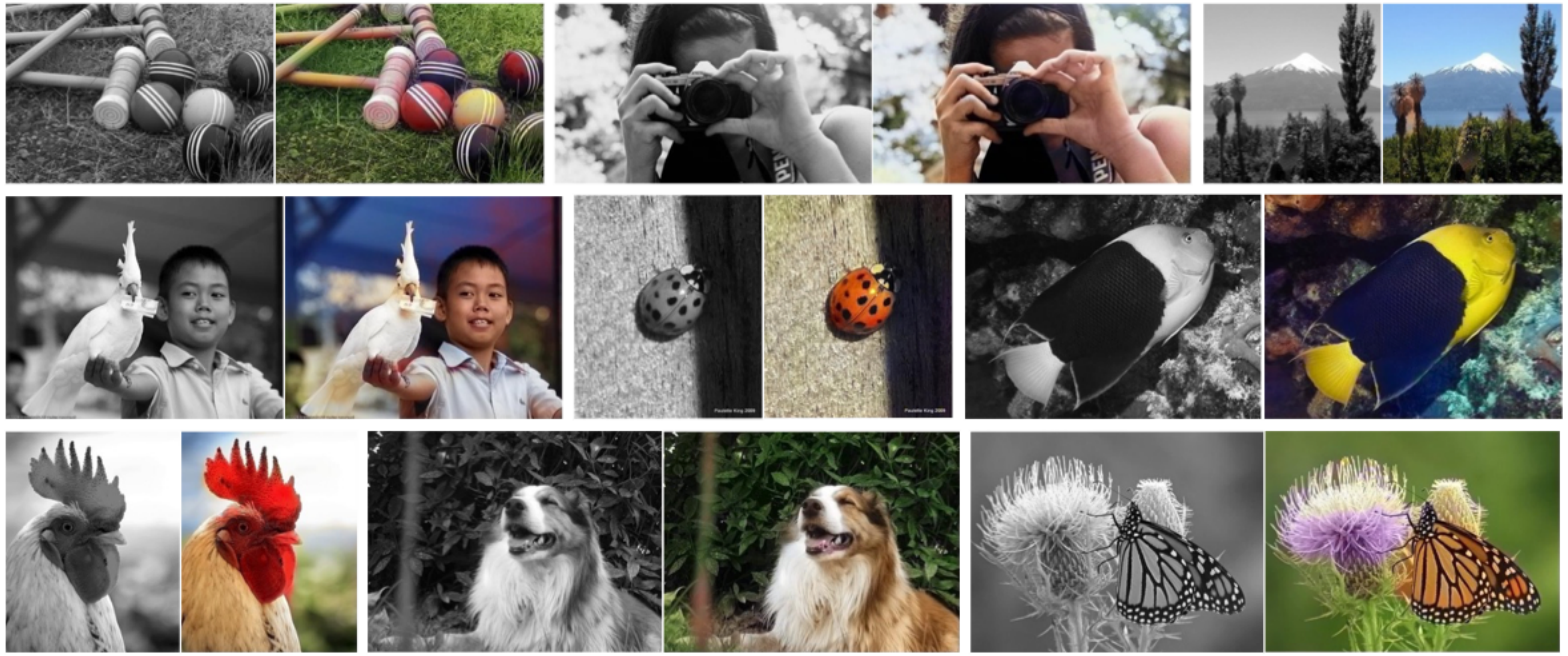
[Pathak et al. (2016)]

Combined L2 + GAN loss

Input Image L2 Loss Adversarial Loss Joint Loss

[Pathak et al. (2016)]

Image colorization

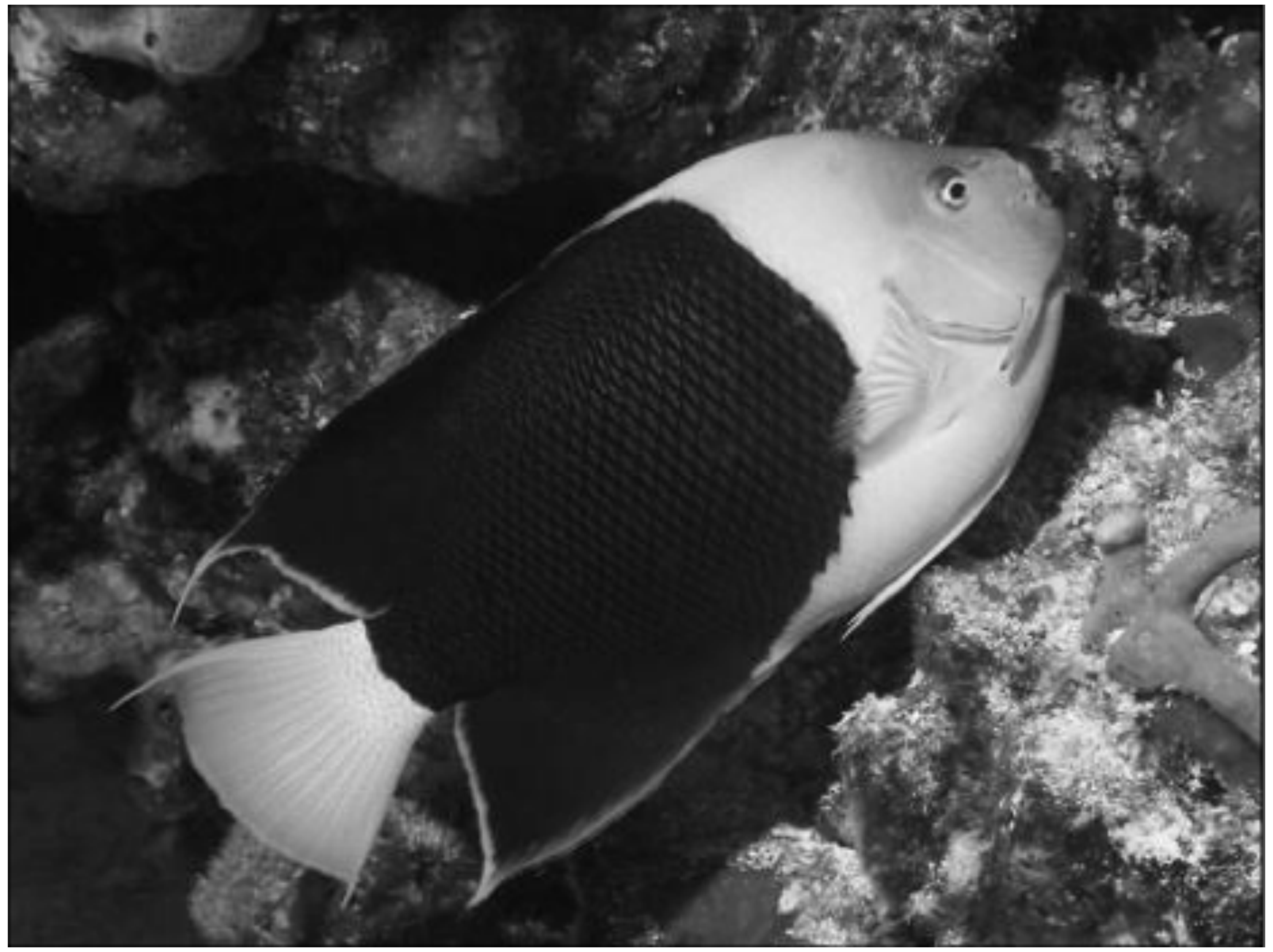


Colorful Image Colorization

Richard Zhang, Phillip Isola, Alexei (Alyosha) Efros

slides from Zhang

http://richzhang.github.io/colorization/



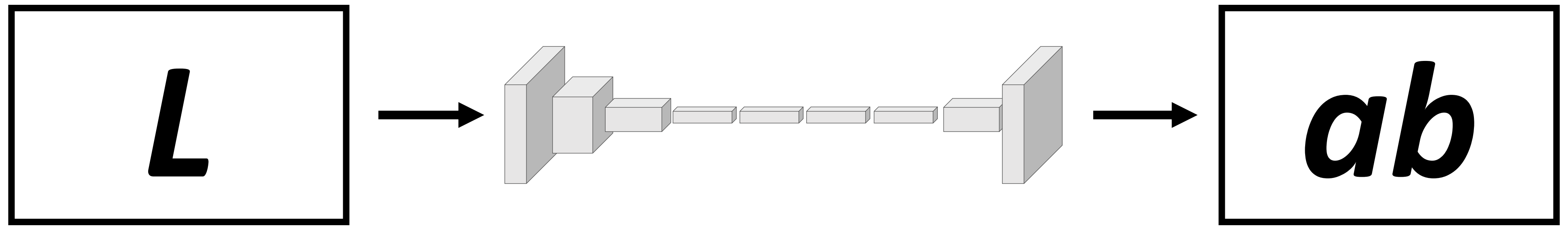
$\mathcal{F}$

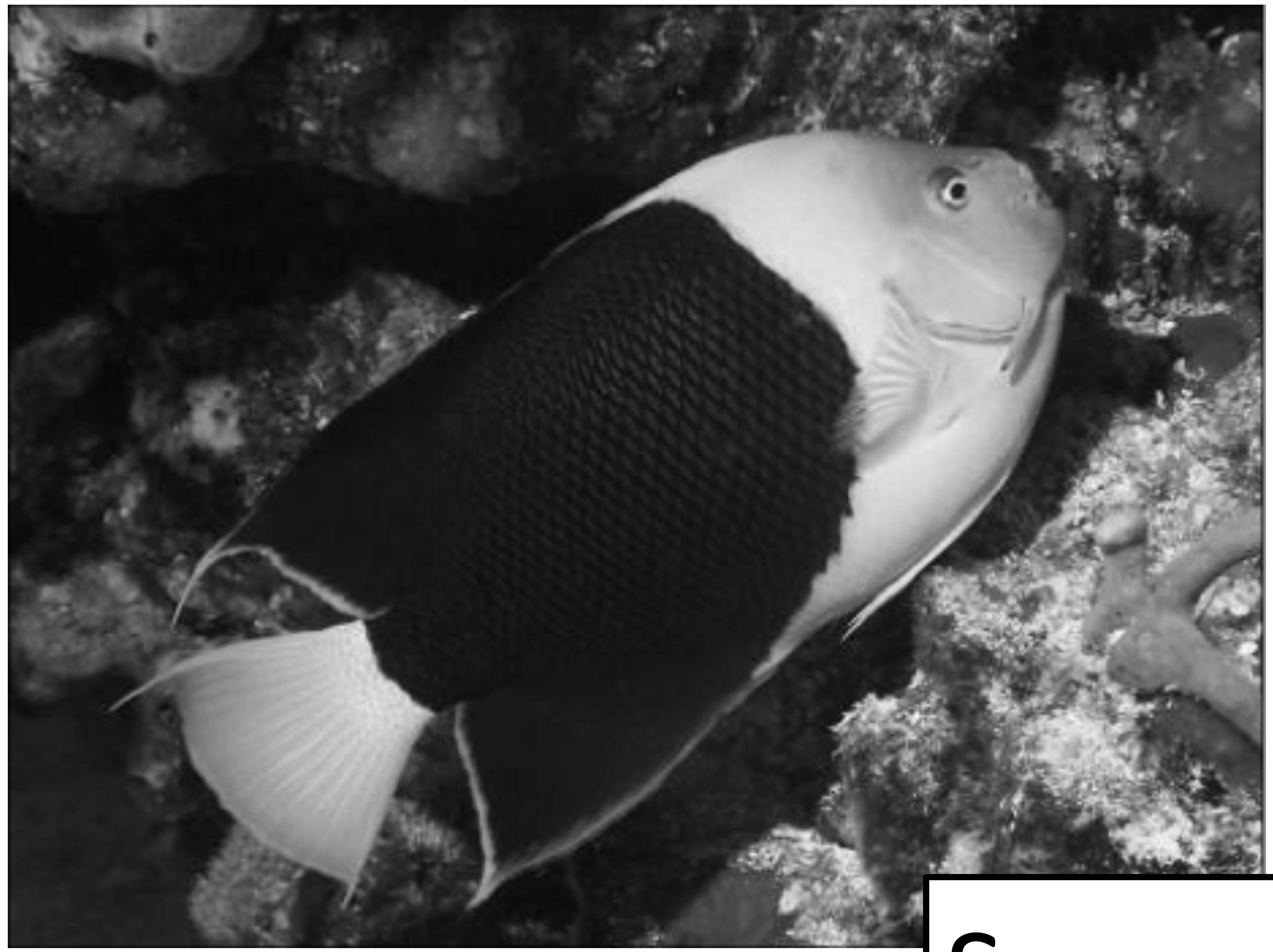
Grayscale image: *L* channel

$\mathbf{X} \in \mathbb{R}^{H \times W \times 1}$

Color information: *ab* channels

$\widehat{\mathbf{Y}} \in \mathbb{R}^{H \times W \times 2}$





$\mathcal{F}$

Grayscale image

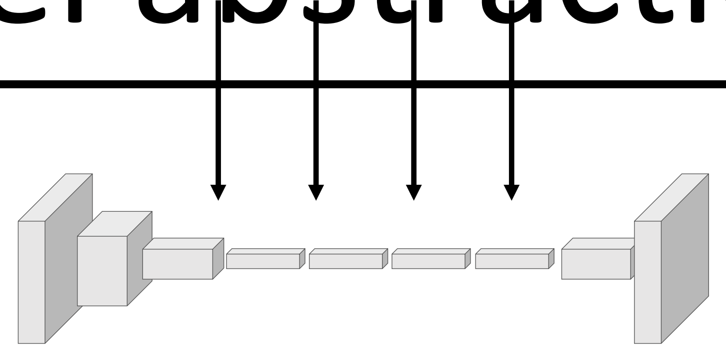
$\mathbf{X} \in \mathbb{R}^{H \times W \times}$

Semantics? Higher-level abstraction?

oncatenate (L,ab)

$(\mathbf{X}, \widehat{\mathbf{Y}})$

L



ab

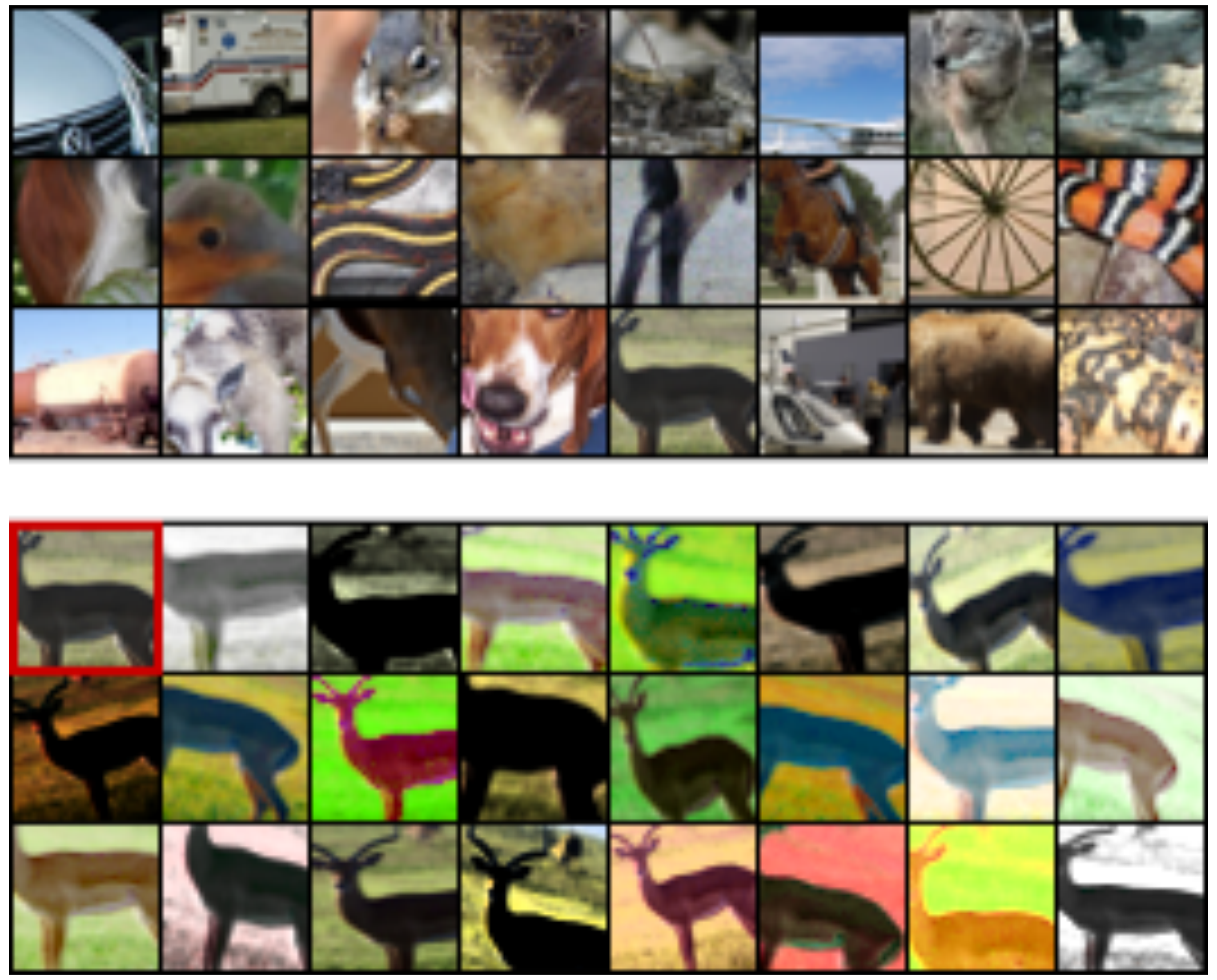
"Free" supervisory signal

From Images: Predicting Rotations

Which image has the correct rotation?

Predicting Image Rotations - Gidaris et al., 2018

[Dosovitskiy et al. ICLR 2014]

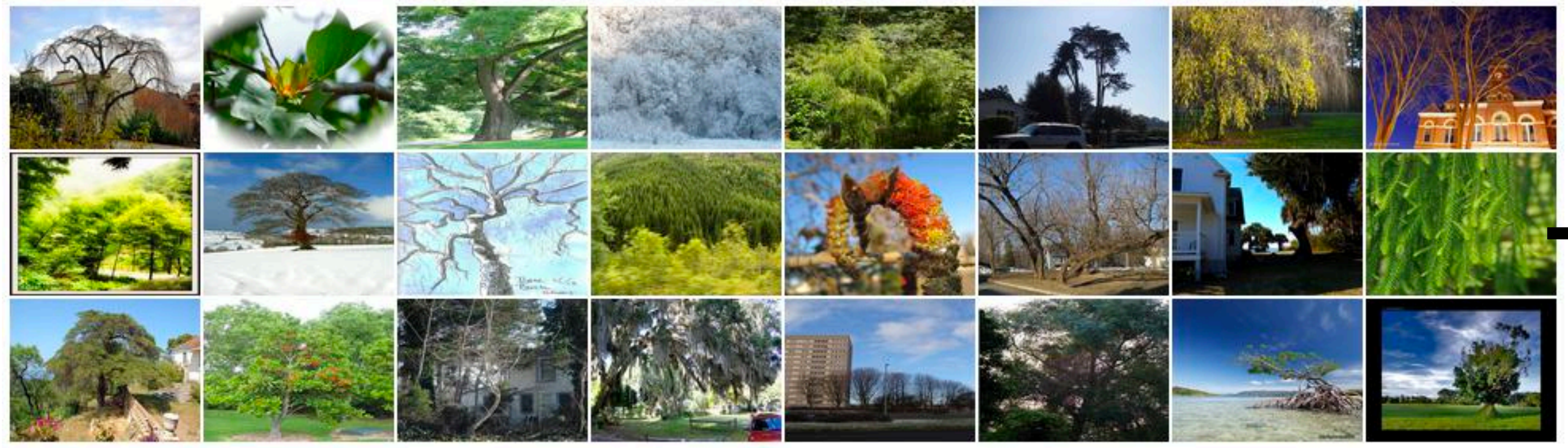


- 1 class = single image + its transformations
- Learn to classify each "class"
- Domain knowledge about appropriate transformations
- does not scale

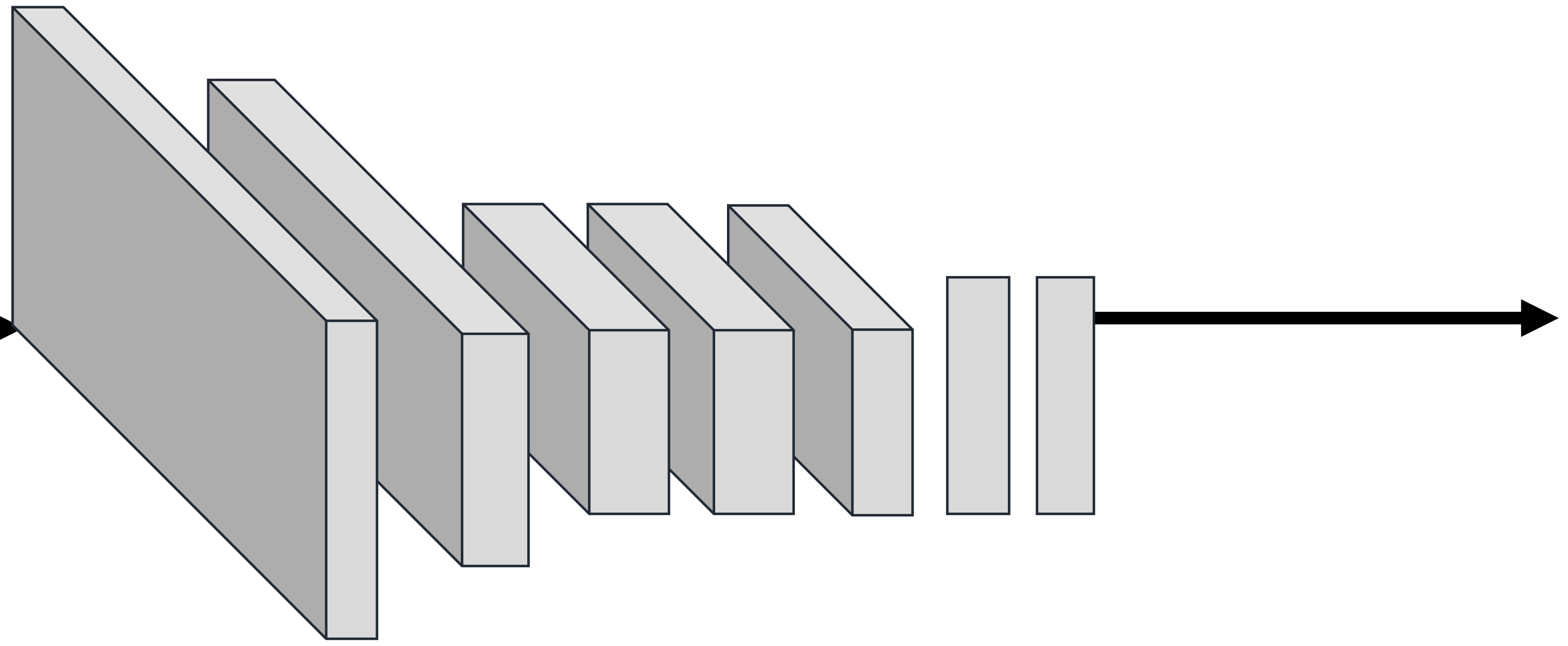
Many different self-supervision tasks, how to evaluate?

Context auto encoders - Pathak et al., 2016

Self-supervised pre-training



Pre-train data

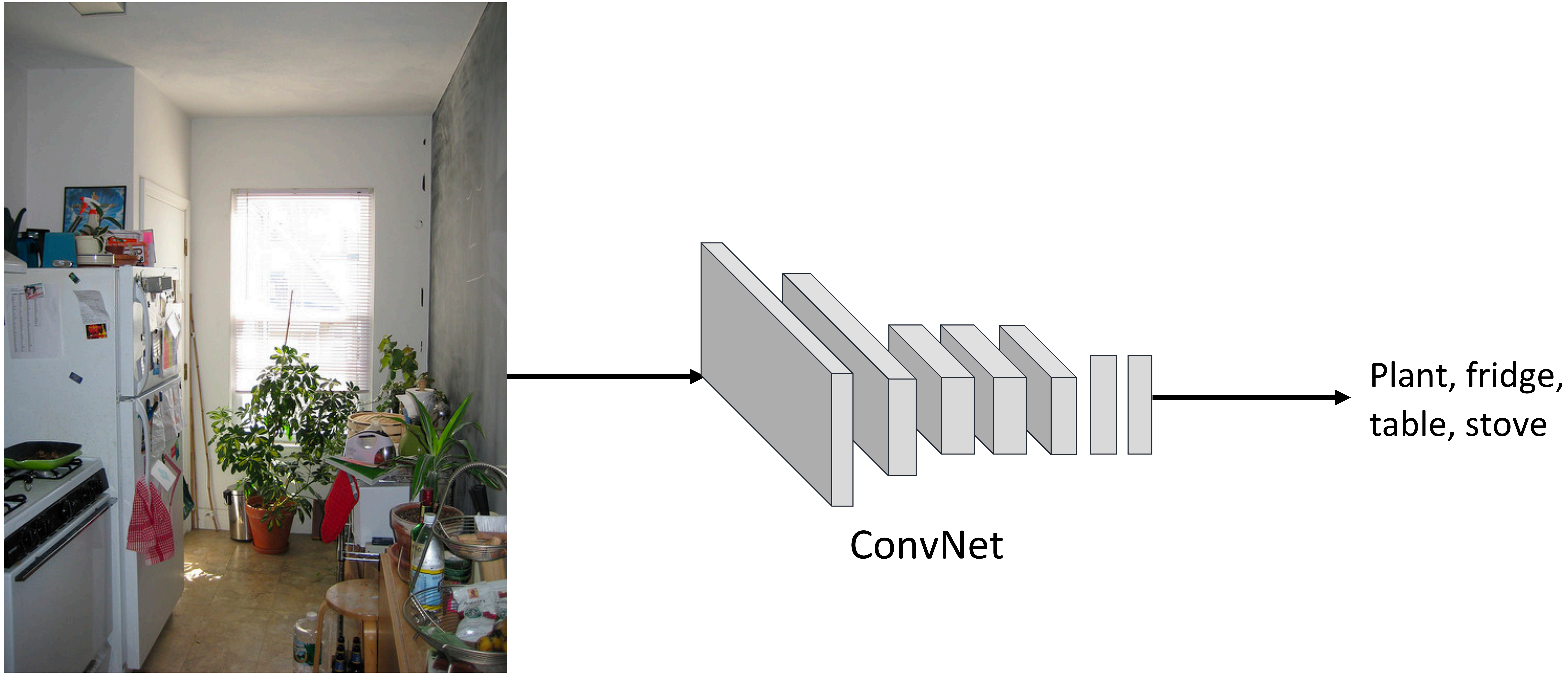


ConvNet

Position/Colorization

Learn a representation

Fine-tune on end task (Image Classification)



Tests representation as well as how good the pre-training initialization is

Fine-tune on end task

Initialization (ResNet101)	End task	
	ImageNet top-5 accuracy	VOC07 Detection mAP
ImageNet Supervised	85.1	74.2
Relative Position	59.2	66.8
Colorization	62.5	65.5

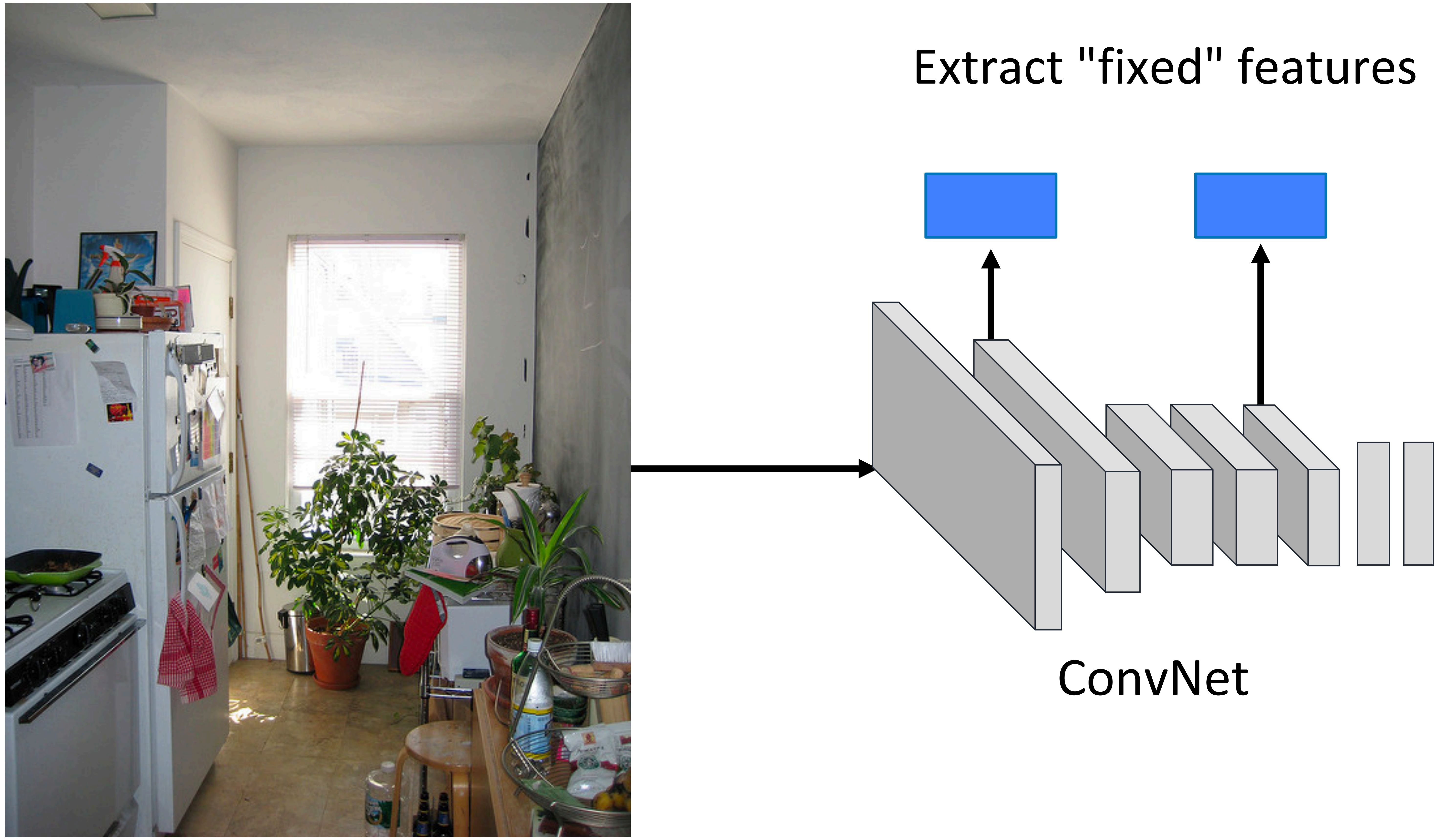
- Multi-task self-supervised visual learning, C Doersch, A Zisserman, ICCV 2017

Are they complementary?

Initialization (ResNet101)	End task	
	ImageNet top-5 accuracy	VOC07 Detection mAP
ImageNet Supervised	85.1	74.2
Relative Position	59.2	66.8
Colorization	62.5	65.5
Relative Position + Colorization (Multi-task)	66.6	68.8

- Multi-task self-supervised visual learning, C Doersch, A Zisserman, ICCV 2017

Train linear classifiers on "fixed" features



Tests how good the representation is (linearly separable)

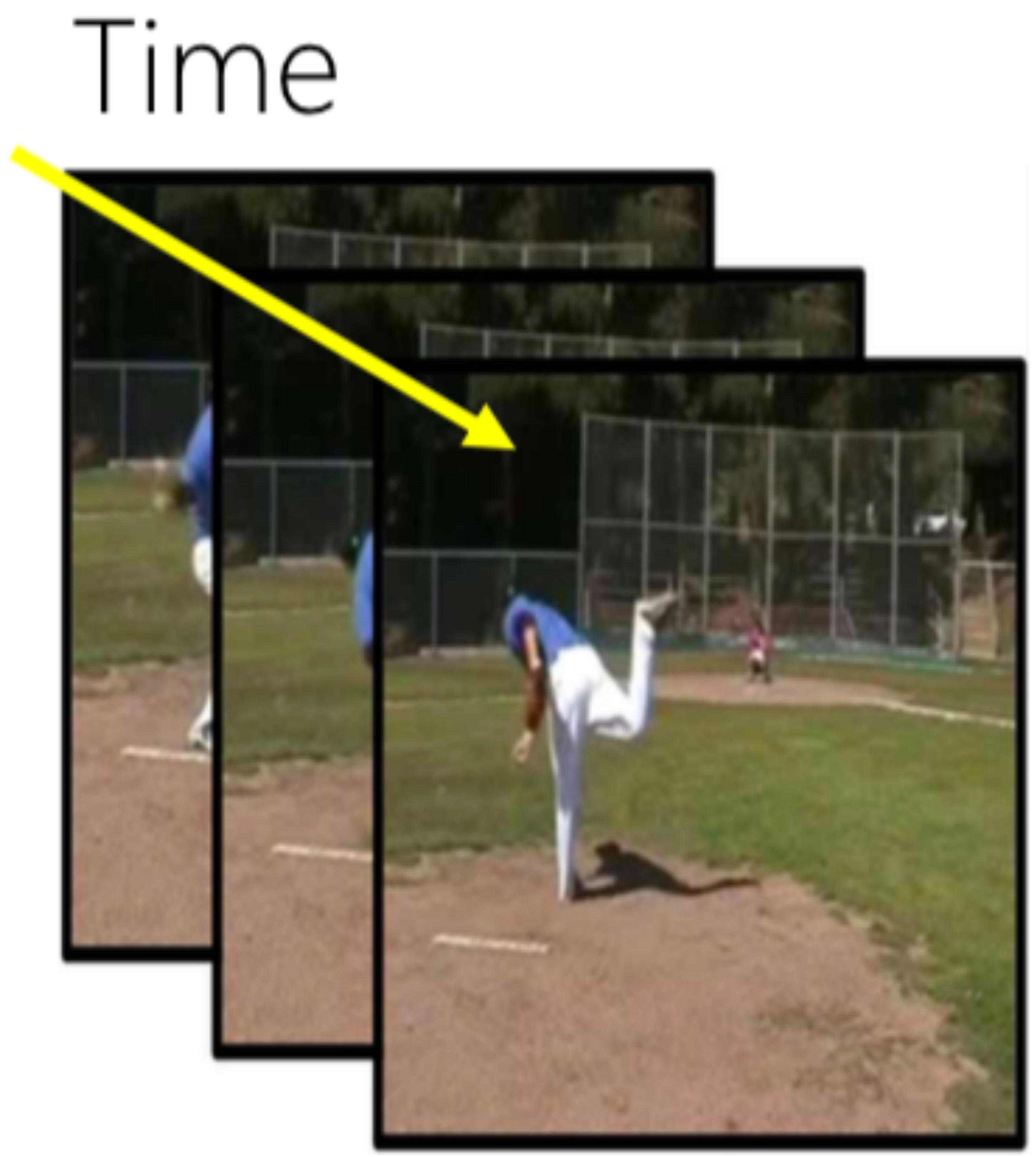
Self-supervision in computer vision

- Using images
- Using video
- Using video and sound

Video

- Video is a "sequence" of frames
- How to get "self-supervision"?

- Predict order of frames
- Fill in the blanks
- Track objects and predict their position



“Sequence” of data

Video

- Slow feature
  - Neighborhood frames should have similar features

$$\mathcal{U}_2 = \{\langle (j,k), p_{jk}\rangle : \boldsymbol{x}_j, \boldsymbol{x}_k \in \mathcal{U} \text{ and } p_{jk} = \mathbb{1}(0 \leq j-k \leq T)\},$$

$$\begin{aligned} R_2(\boldsymbol{\theta}, \mathcal{U}) &= \sum_{(j,k)\in\mathcal{U}_2} D_\delta(\mathbf{z}_{\boldsymbol{\theta}}(\boldsymbol{x}_j), \mathbf{z}_{\boldsymbol{\theta}}(\boldsymbol{x}_k), p_{jk}) \\ &= \sum_{(j,k)\in\mathcal{U}_2} p_{jk}\, d(\mathbf{z}_{\boldsymbol{\theta}j}, \mathbf{z}_{\boldsymbol{\theta}k}) + \overline{p_{jk}}\, \max(\delta - d(\mathbf{z}_{\boldsymbol{\theta}j}, \mathbf{z}_{\boldsymbol{\theta}k}), 0), \end{aligned}$$

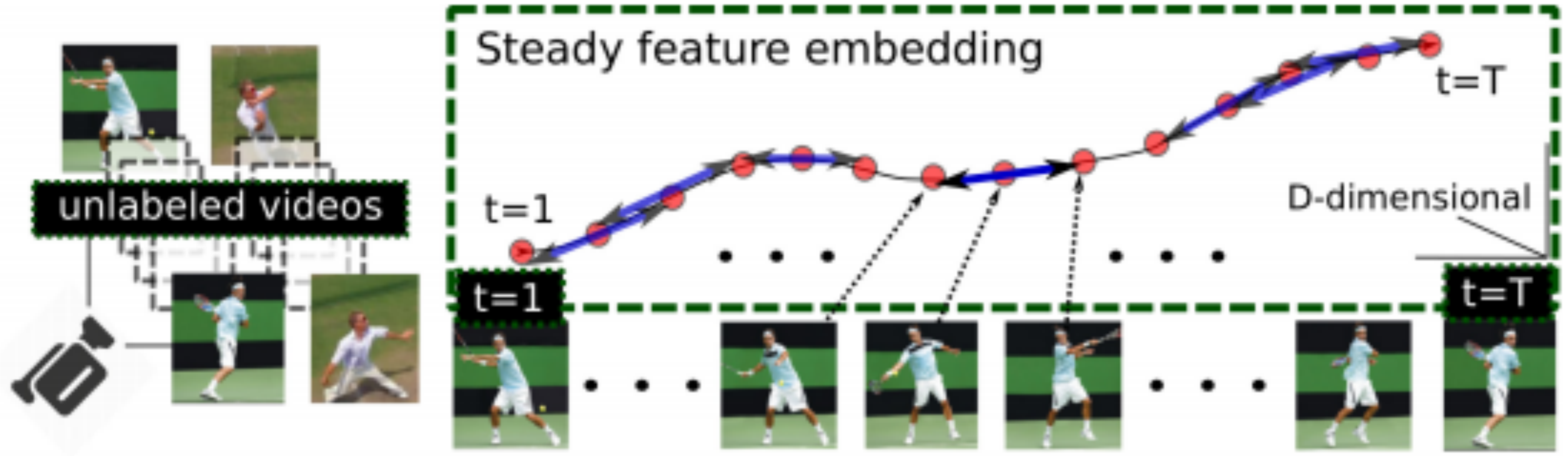
Mobahi, H., Collobert, R., & Weston, J. Deep learning from temporal coherence in video. In *ICML 2009*.

Wiskott, L., & Sejnowski, T. J. (2002). Slow feature analysis: Unsupervised learning of invariances. *Neural computation*, *14*(4), 715-770.

[A Survey to Self-Supervised Learning, Naiyan Wang]

Video

- Slow and steady feature
  - Not only similar, but also smooth
  - Extend to triplet setting (Not triplet loss!)

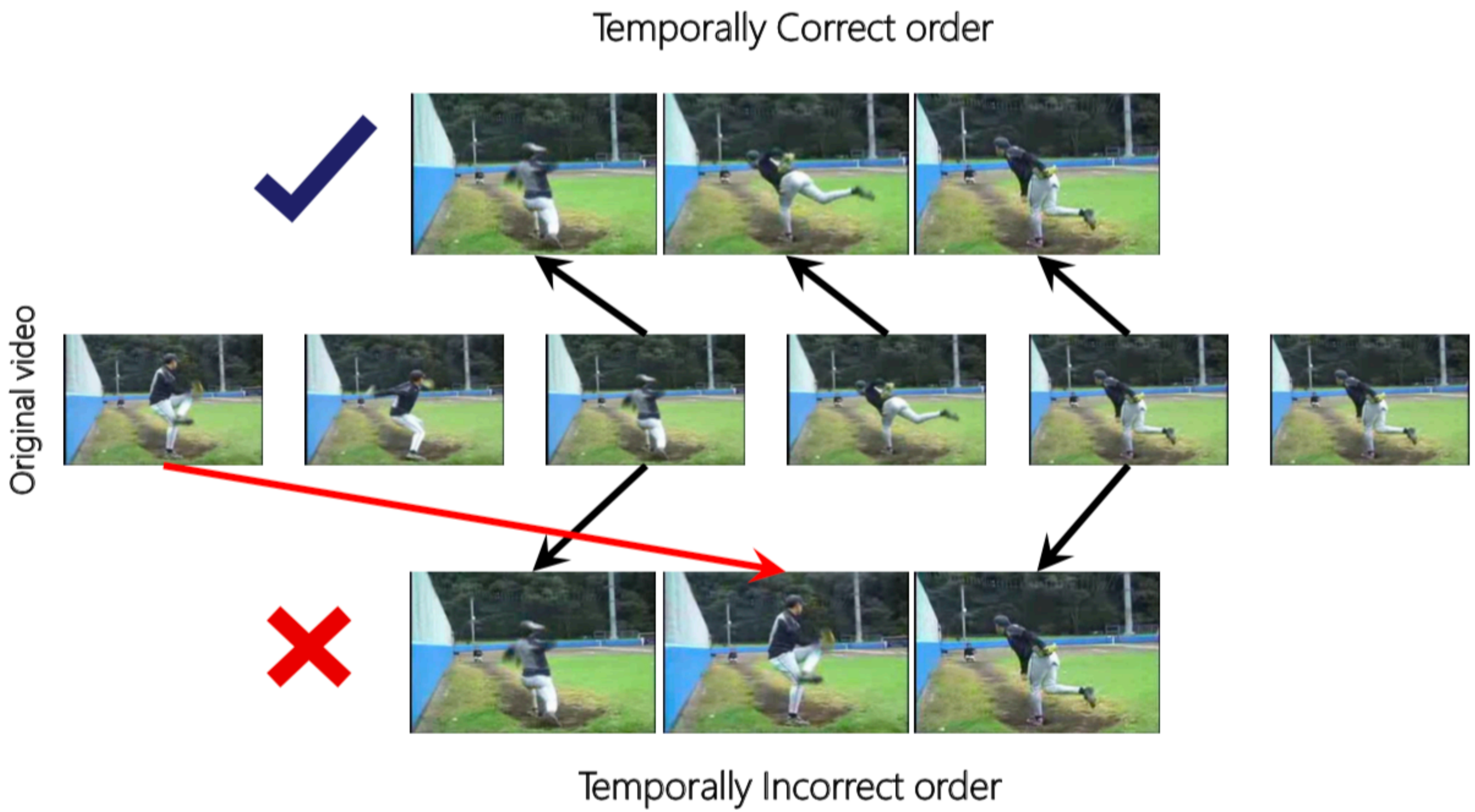


$$\mathcal{U}_3 = \{\langle (l,m,n), p_{lmn} \rangle : \boldsymbol{x}_l, \boldsymbol{x}_m, \boldsymbol{x}_n \in \mathcal{U} \text{ and } p_{lmn} = \mathbb{1}(0 \leq m - l = n - m \leq T)\}.$$

$$R_3(\boldsymbol{\theta}, \mathcal{U}) = \sum_{(l,m,n) \in \mathcal{U}_3} D_\delta(\mathbf{z}_{\boldsymbol{\theta}l} - \mathbf{z}_{\boldsymbol{\theta}m},\ \mathbf{z}_{\boldsymbol{\theta}m} - \mathbf{z}_{\boldsymbol{\theta}n},\ p_{lmn}),$$

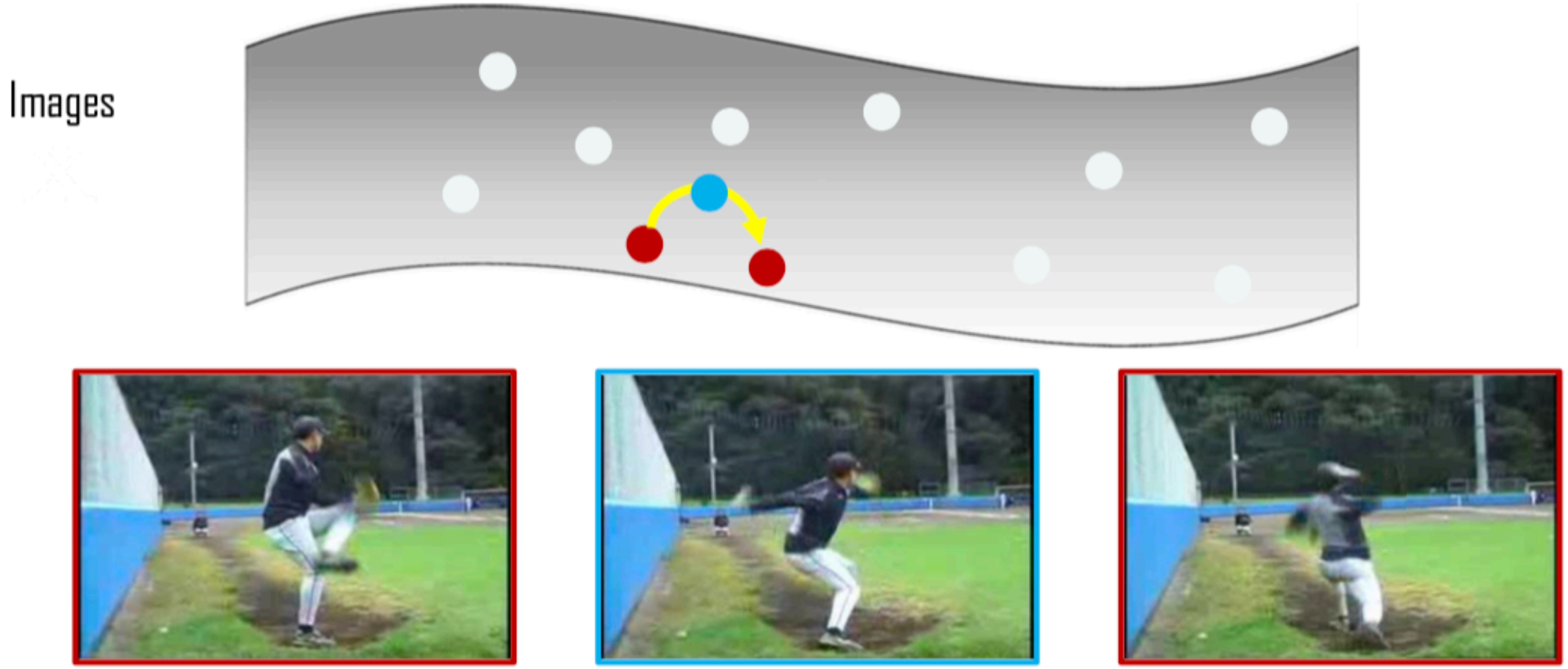
Jayaraman, D., & Grauman, K. Slow and steady feature analysis: higher order temporal coherence in video. In *CVPR 2016*.

From video: shuffle and learn



Shuffle and Learn – I. Misra, L. Zitnick, M. Hebert – ECCV 2016

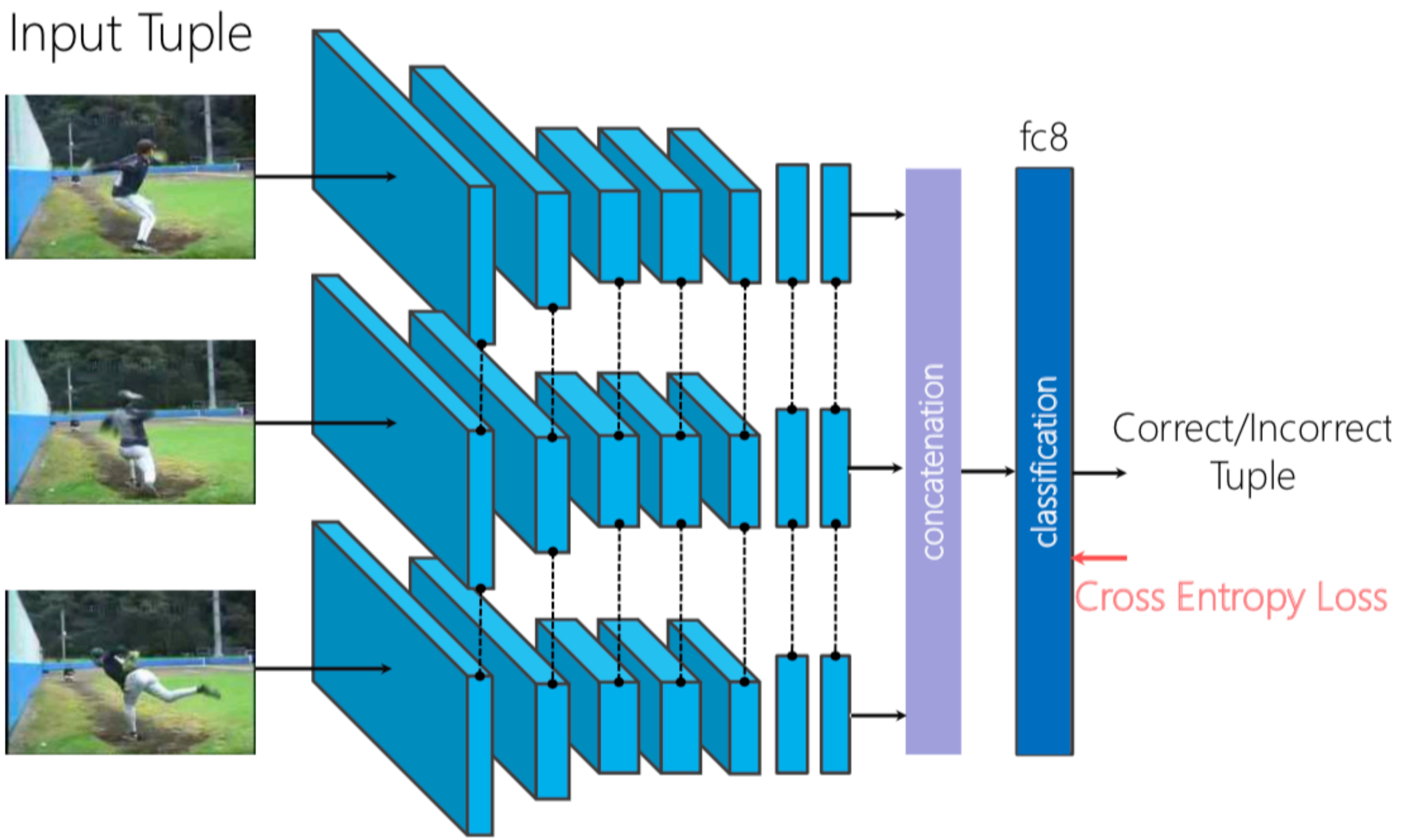
From video: shuffle and learn



Given a start and an end, can this point lie in between?

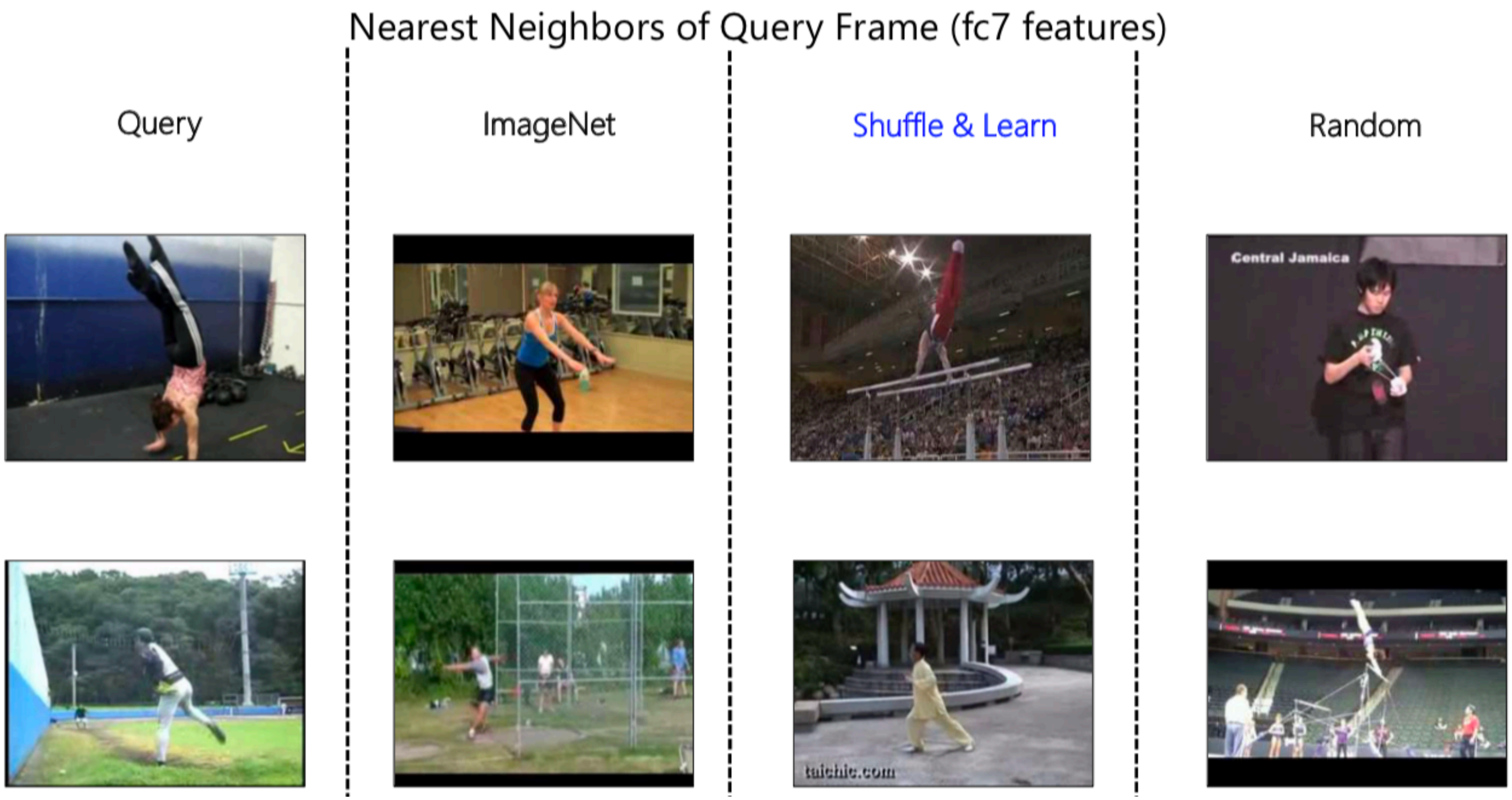
Shuffle and Learn – I. Misra, L. Zitnick, M. Hebert – ECCV 2016

From video: shuffle and learn



Shuffle and Learn – I. Misra, L. Zitnick, M. Hebert – ECCV 2016

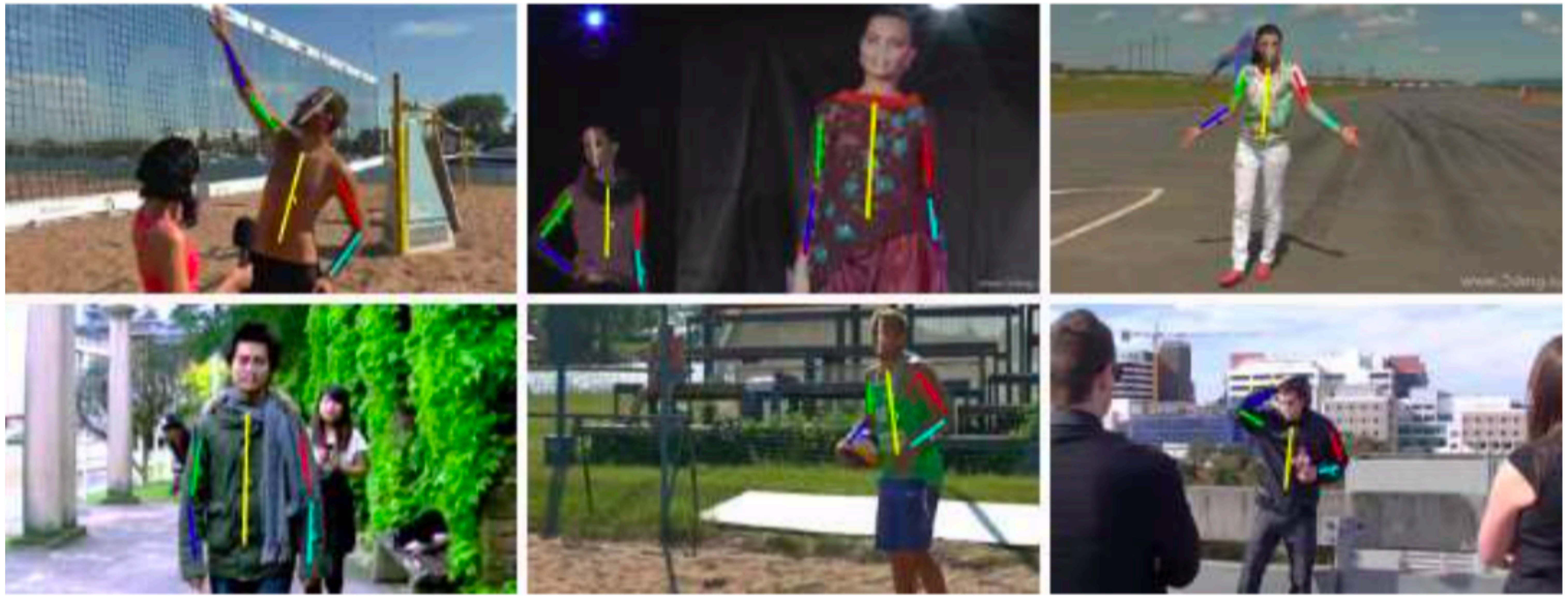
From video: shuffle and learn



Shuffle and Learn – I. Misra, L. Zitnick, M. Hebert – ECCV 2016

From video: shuffle and learn

Fine-tune on Human Keypoint Estimation



Shuffle and Learn – I. Misra, L. Zitnick, M. Hebert – ECCV 2016

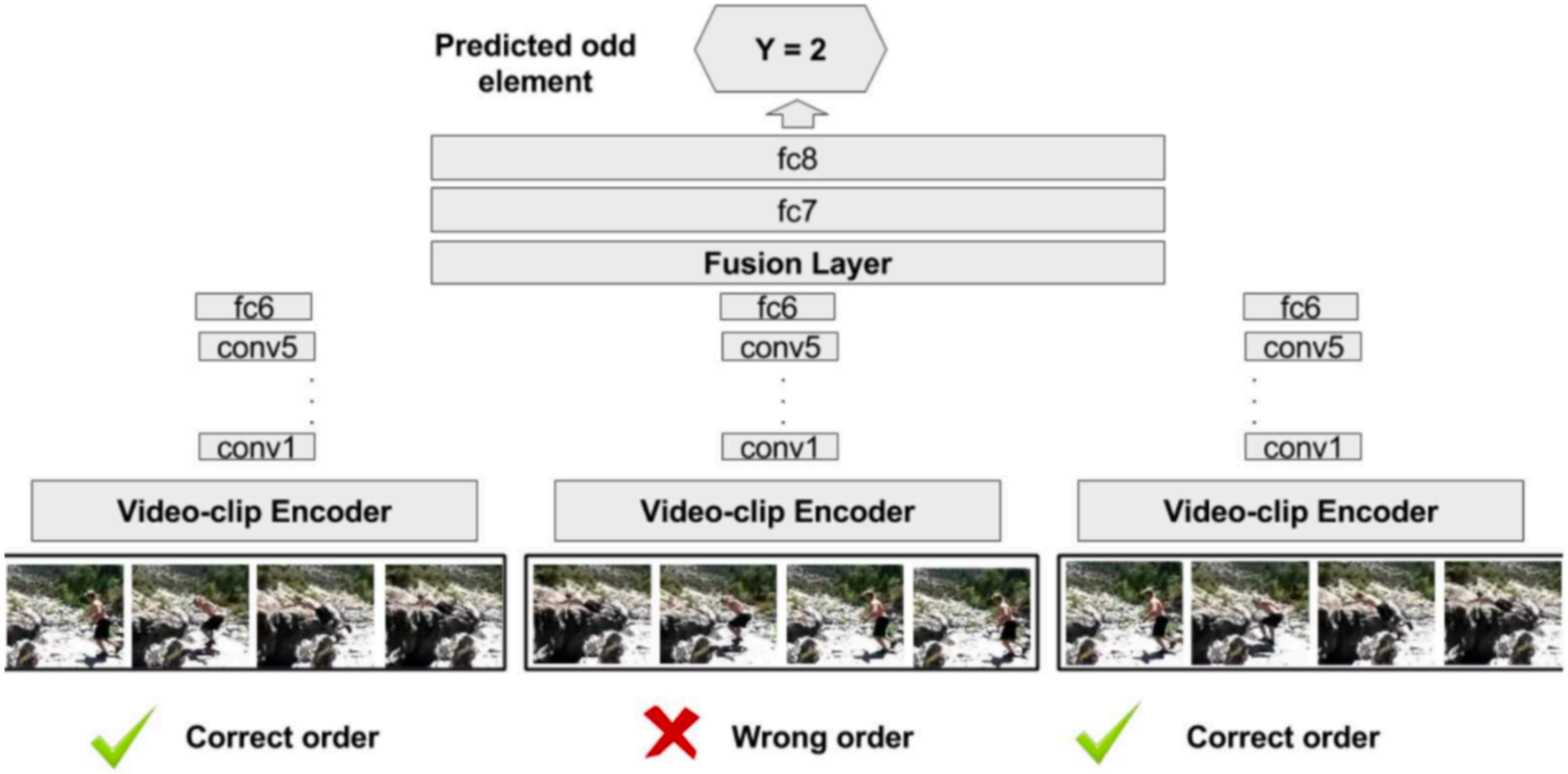
From video: shuffle and learn

Fine-tune on Human Keypoint Estimation

Initialization (AlexNet)	End task	
	FLIC Dataset Keypoints AUC	MPII Dataset Keypoints AUC
ImageNet Supervised	51.3	**47.6**
Shuffle and Learn	**49.6**	47.2

Shuffle and Learn – I. Misra, L. Zitnick, M. Hebert – ECCV 2016

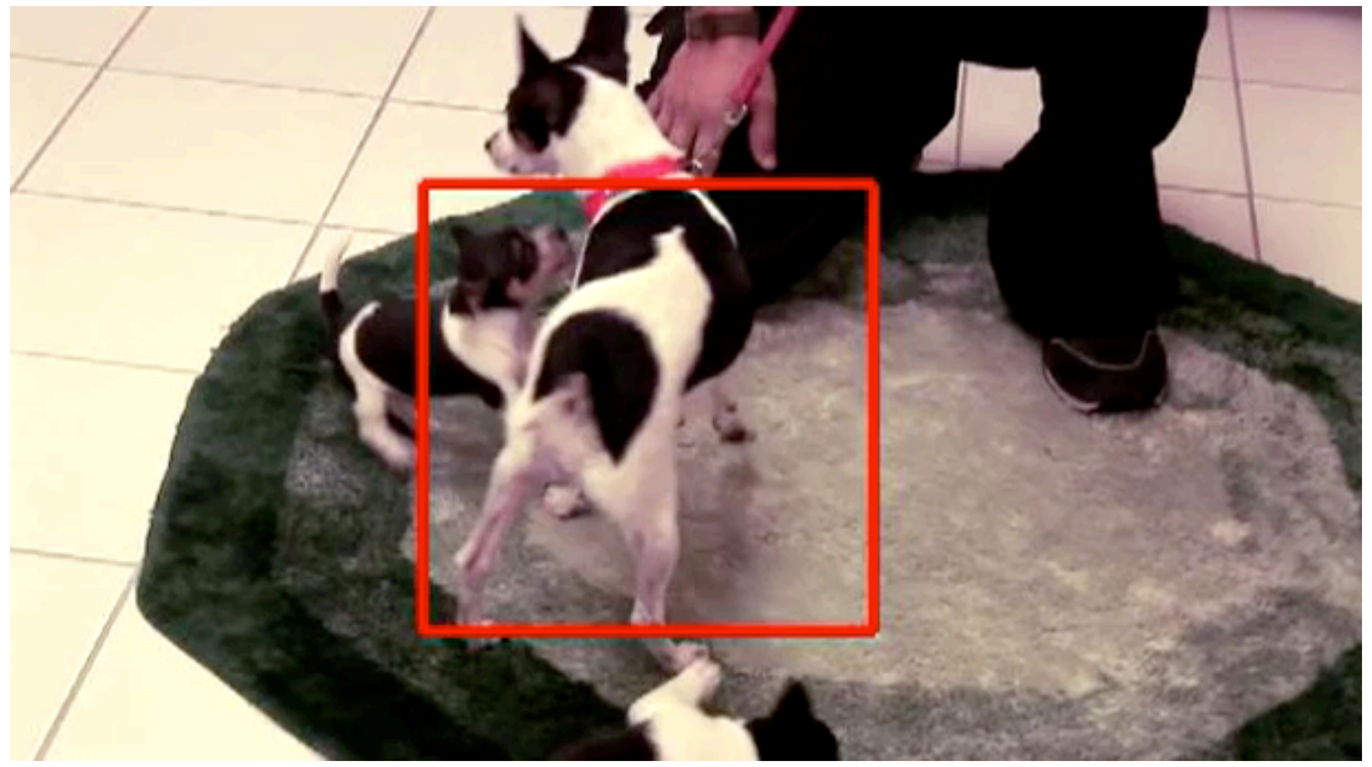
From video: encoding more structure



Odd-one-out networks - Fernando, Bilen, Gavves, Gould in ICCV 2017

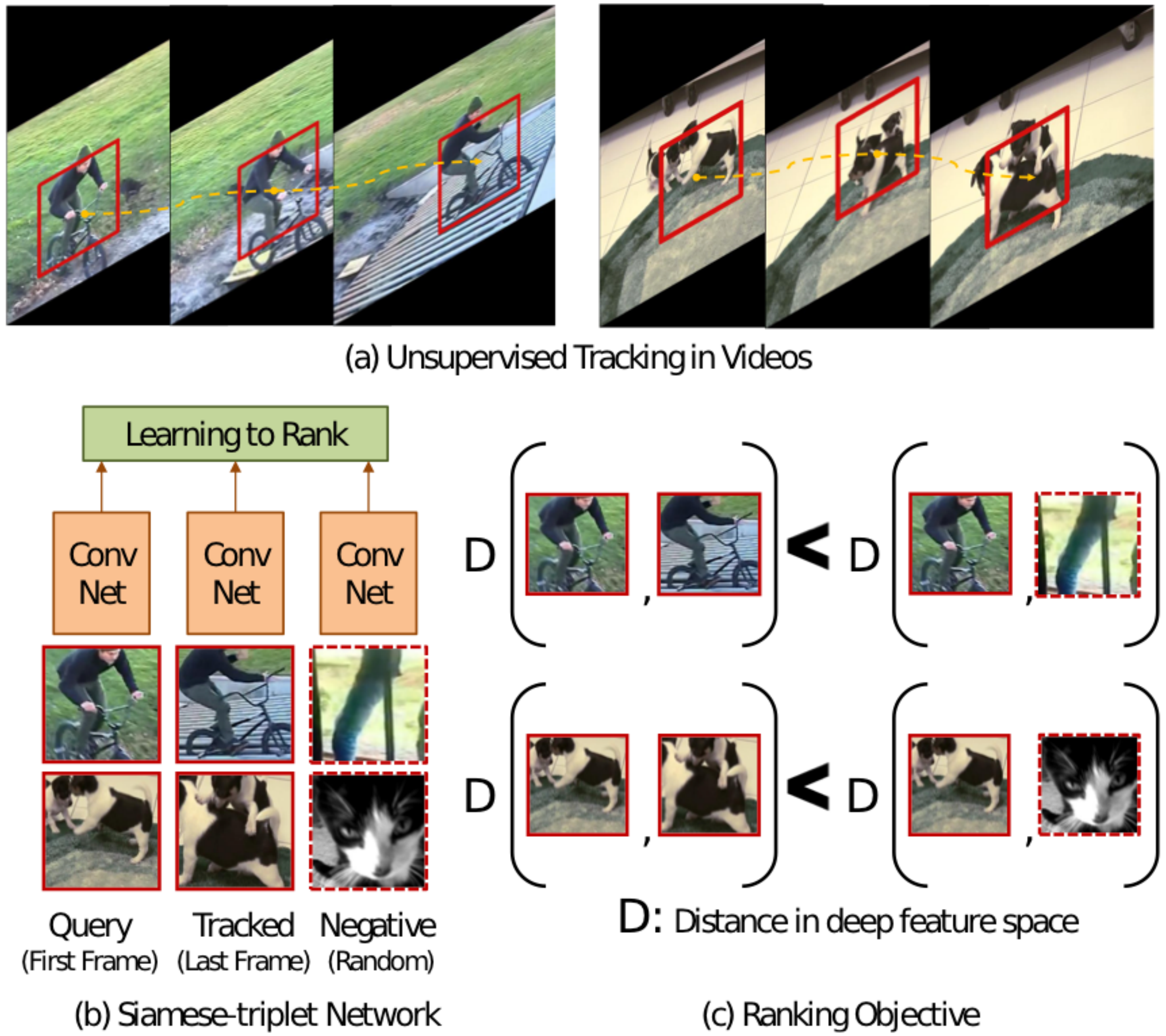
Unsupervised Learning of Visual Representations using Videos, Wang & Gupta 2015

Idea: Object Tracking in Videos



[Wang & Gupta 2015]

Approach

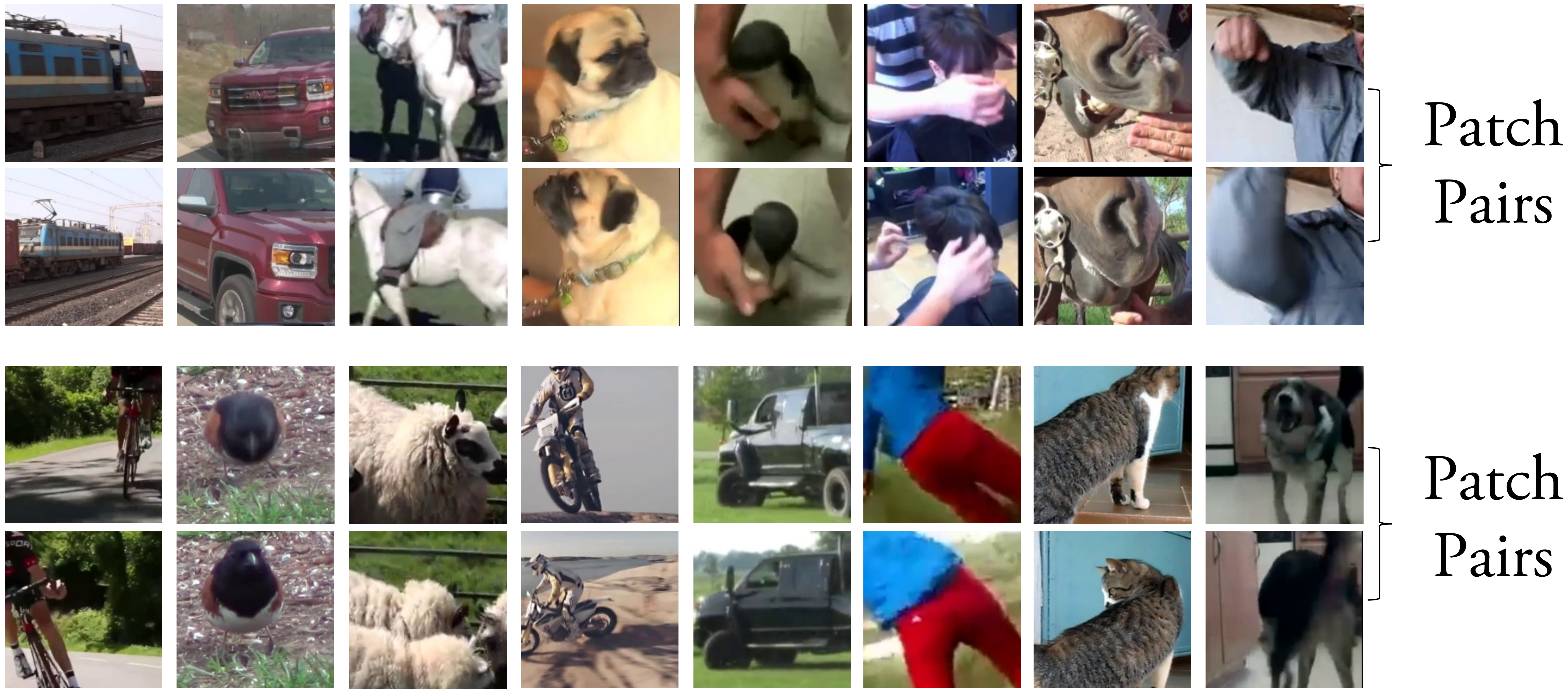


- Use object tracking in videos
- Classify if patches belong to the same track or not

[Wang & Gupta 2015]

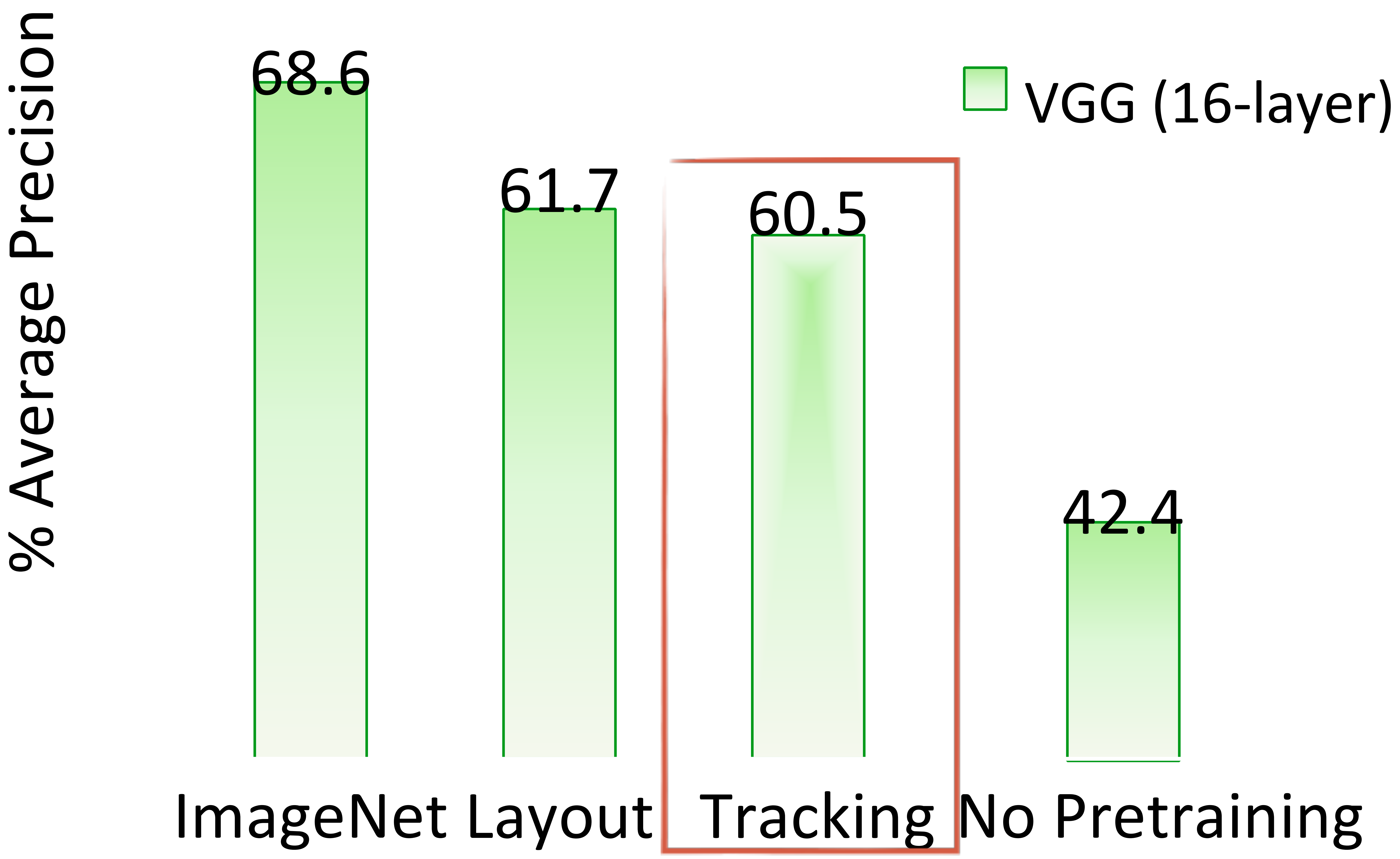
Patch Mining In Videos

- Track 8M patches in 100K videos from YouTube.
- Use off-the-shelf tracking algorithms with no learning.



[Wang & Gupta 2015]

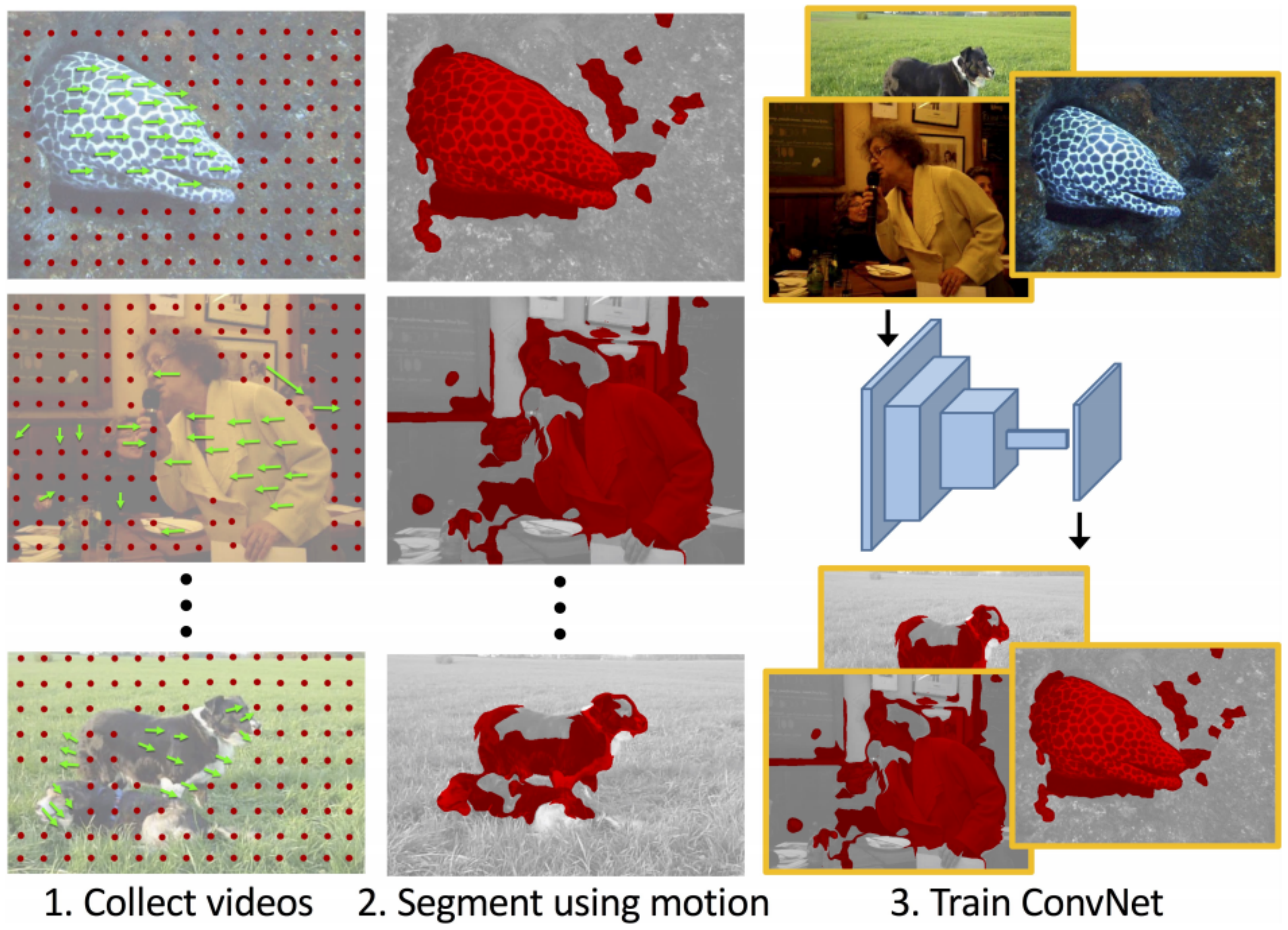
VOC 2007 Detection Performance
(pretraining for R-CNN)



[Wang & Gupta 2015]

Object Movement

- The world is rigid, or at least piecewise rigid
  - Motion provide evidence of how pixels move together
  - The pixels move together are likely to form an object



Pathak, D., Girshick, R., Dollár, P., Darrell, T., & Hariharan, B. Learning Features by Watching Objects Move. *In CVPR 2017*.

Self-supervision in computer vision

- Using images
- Using video
- Using video and sound

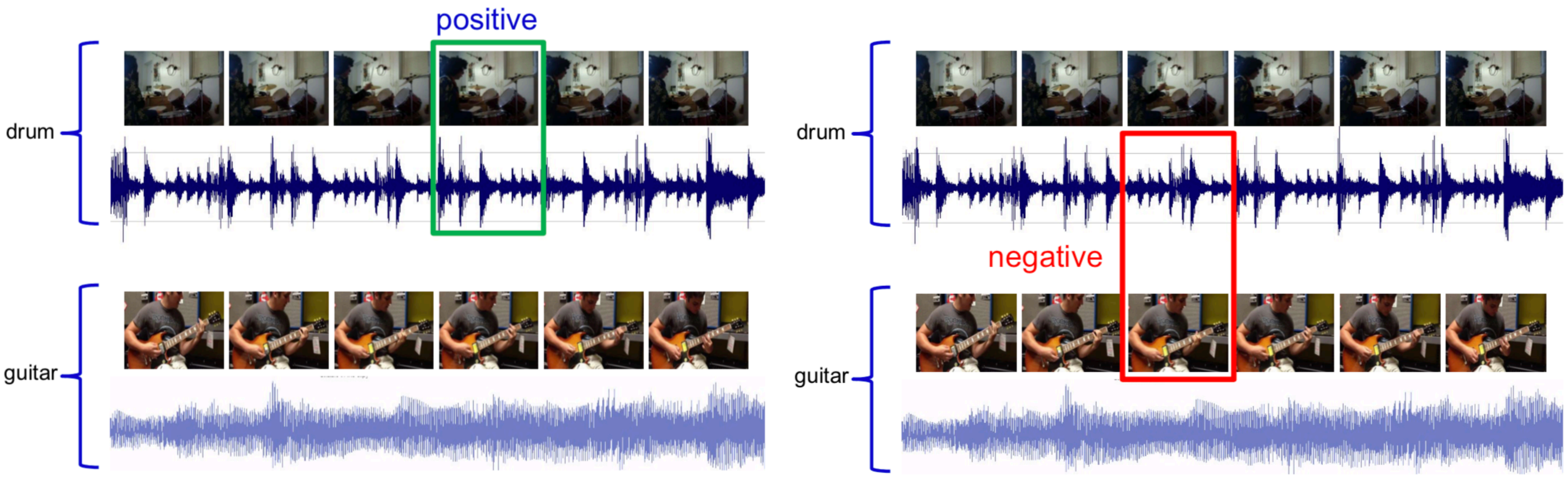
Audio-Visual co-supervision

Train a network to predict if **image** and audio clip correspond

Correspond?

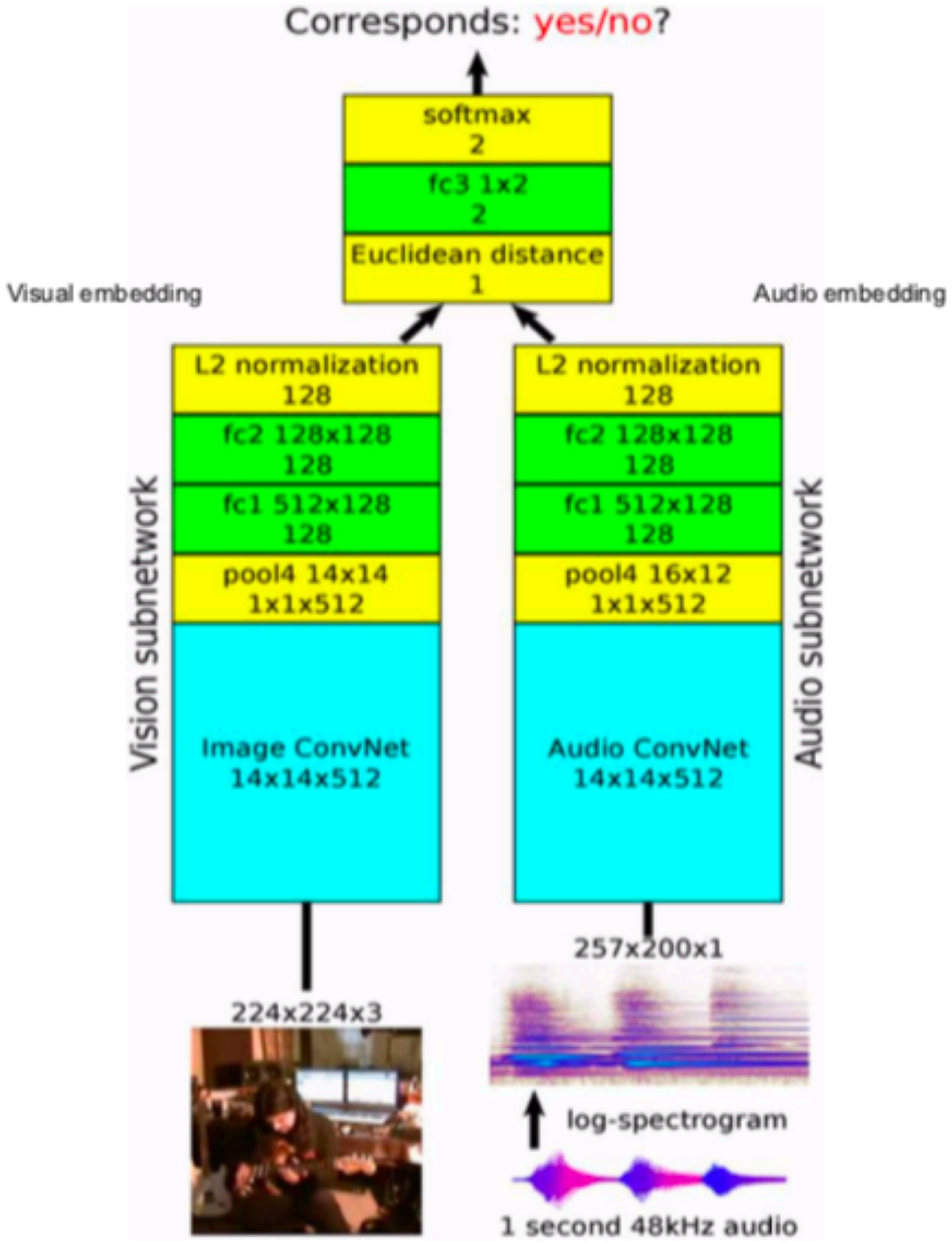
"Objects that Sound", Arandjelović and Zisserman, ICCV 2017 & ECCV 2018

Audio-Visual co-supervision



"Objects that Sound", Arandjelović and Zisserman, ICCV 2017 & ECCV 2018

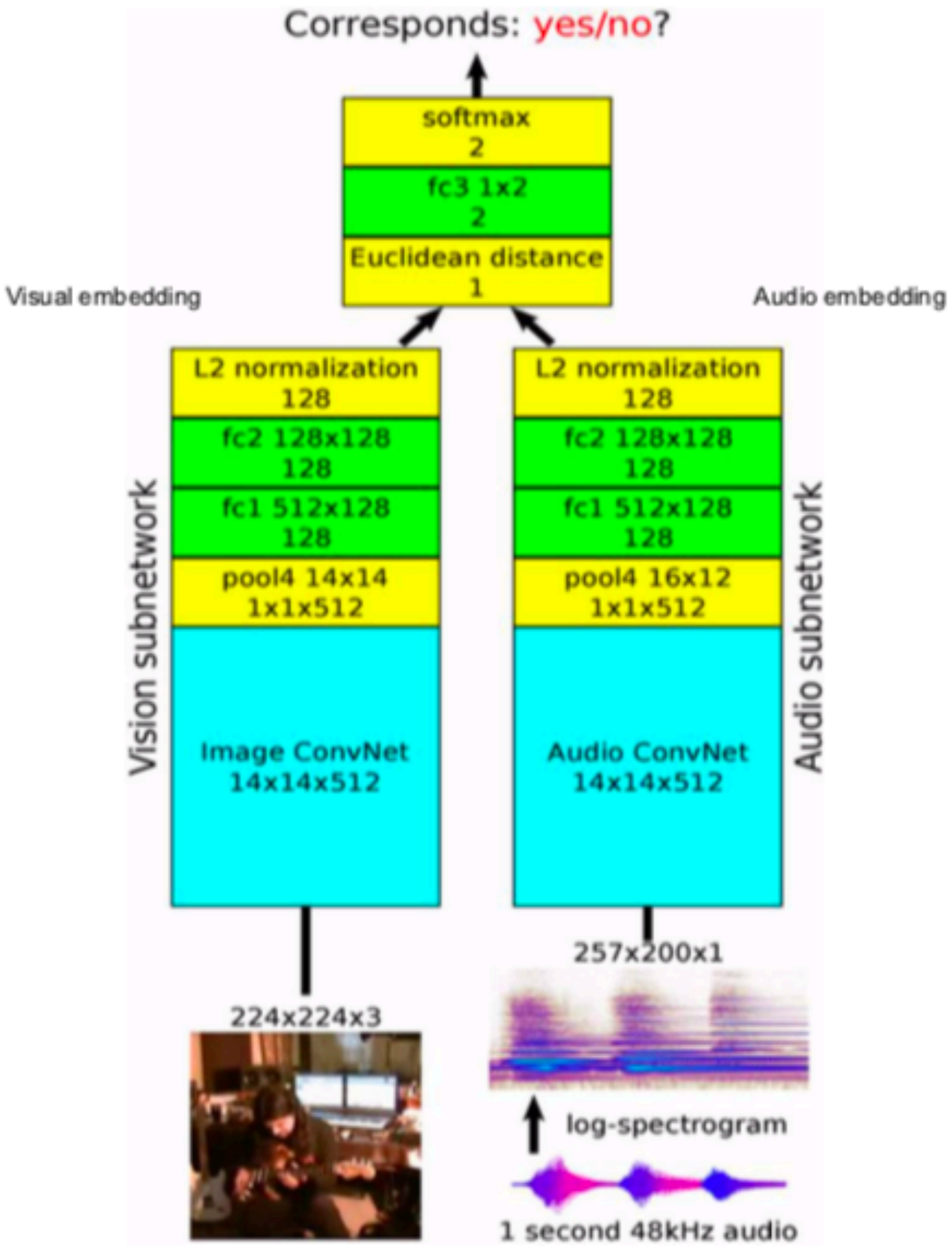
Audio-Visual co-supervision



facebook
Artificial Intelligence Research

"Objects that Sound", Arandjelović and Zisserman, ICCV 2017 & ECCV 2018

Audio-Visual co-supervision



What can be learnt?

- Good representations – Visual features – Audio features
- Intra- and cross-modal retrieval – Aligned audio and visual embeddings
- "What is making the sound?" – Learn to localize objects that sound

"Objects that Sound", Arandjelović and Zisserman, ICCV 2017 & ECCV 2018

Audio-Visual co-supervision

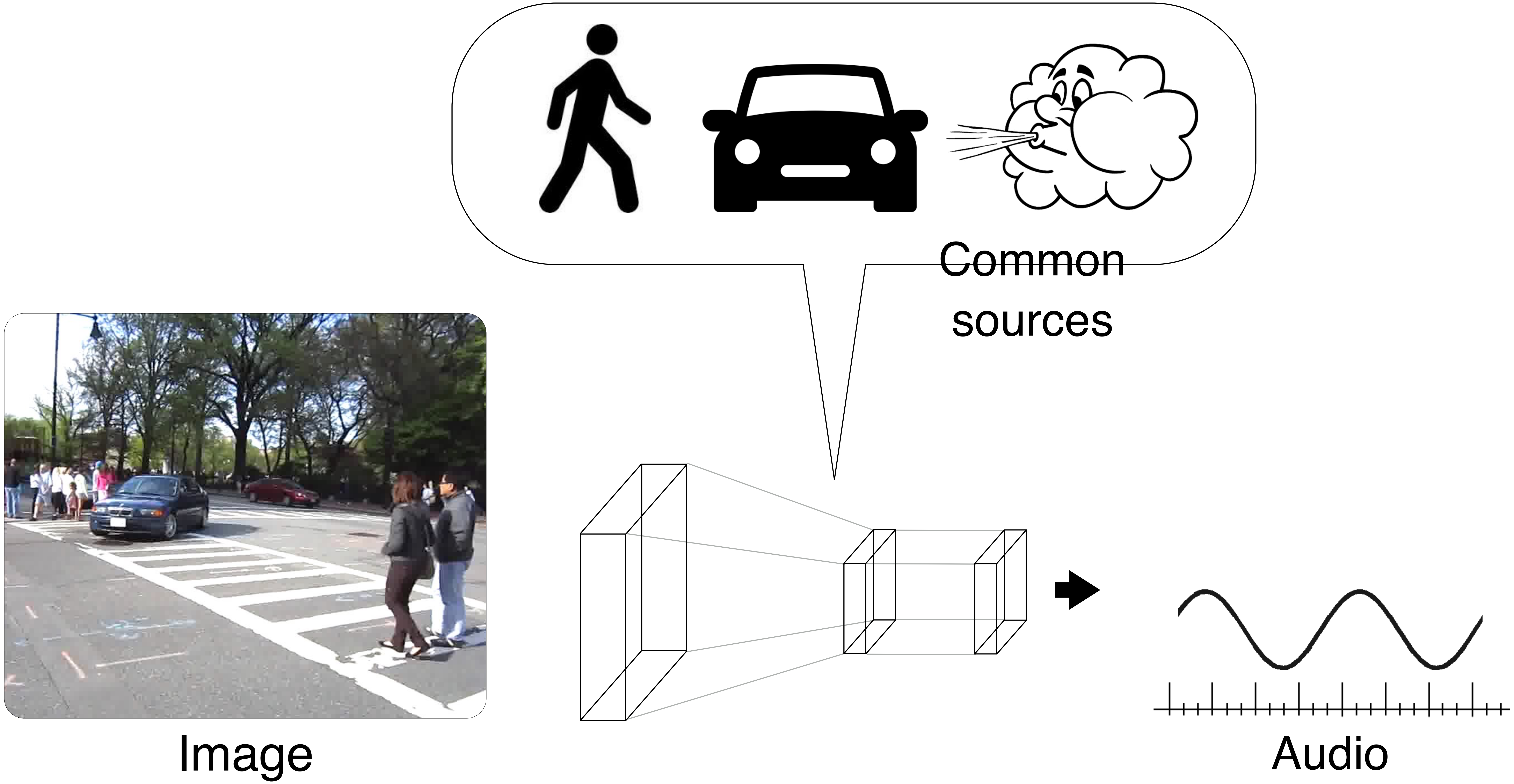
What would make this sound?

Note, no video (motion) information is used

"Objects that Sound", Arandjelović and Zisserman, ICCV 2017 & ECCV 2018

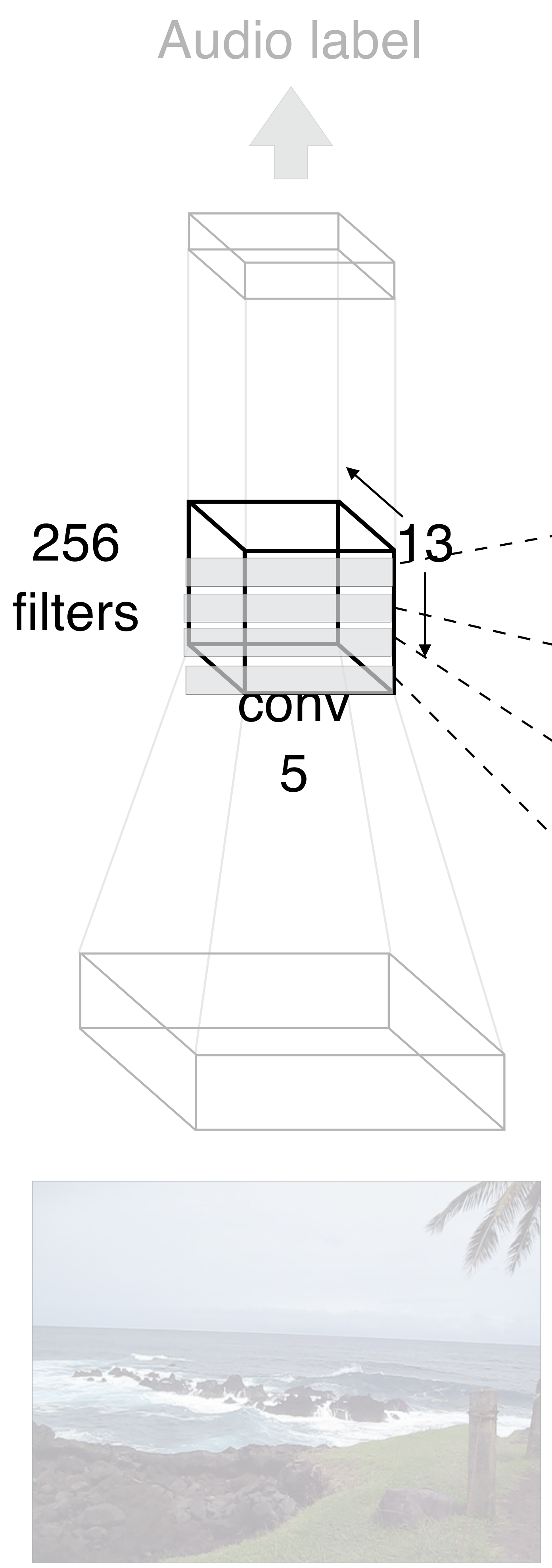
Visual + Audio

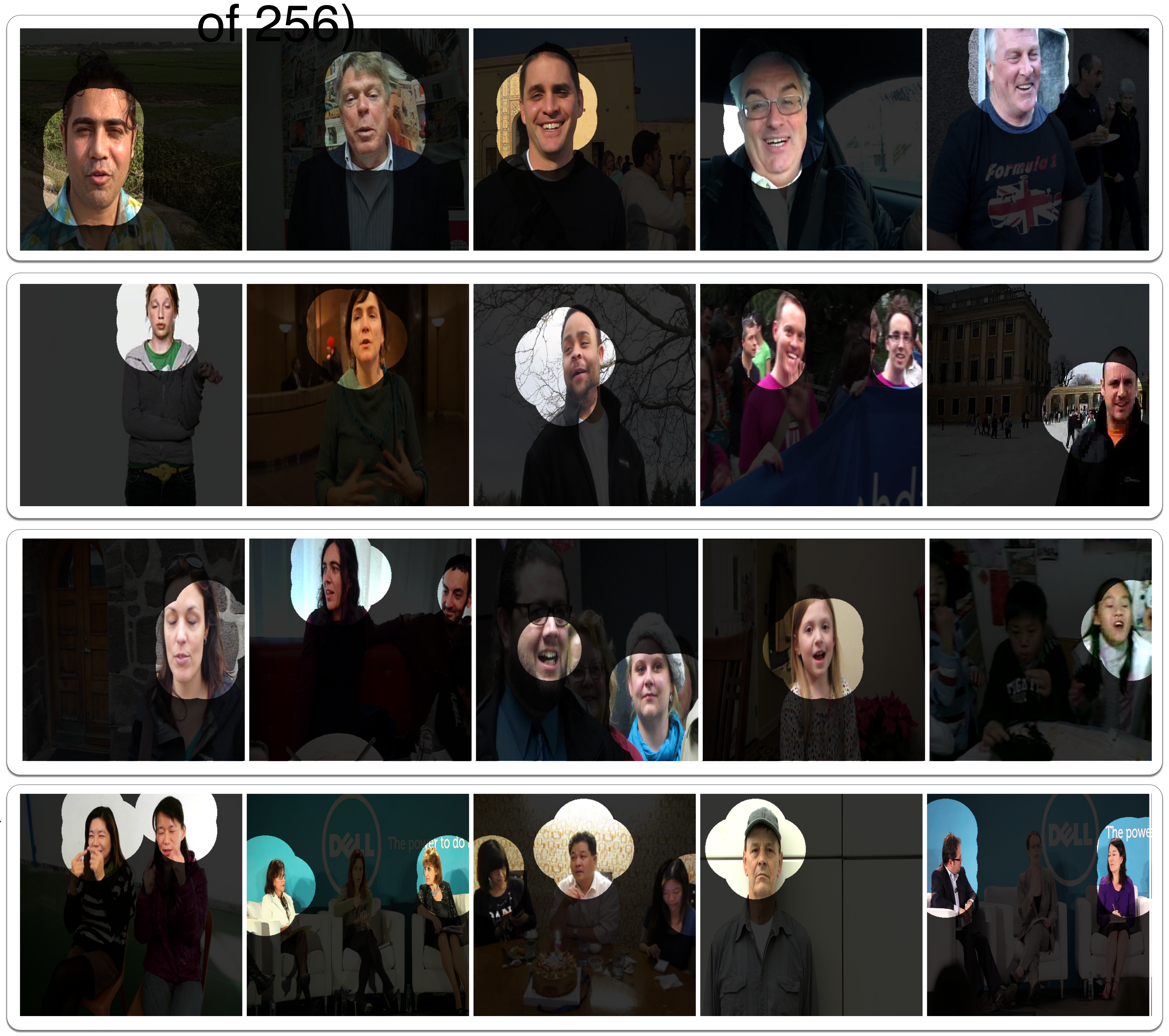
[Ambient Sound Provides Supervision for Visual Learning,
[Owens et al. (2016)]



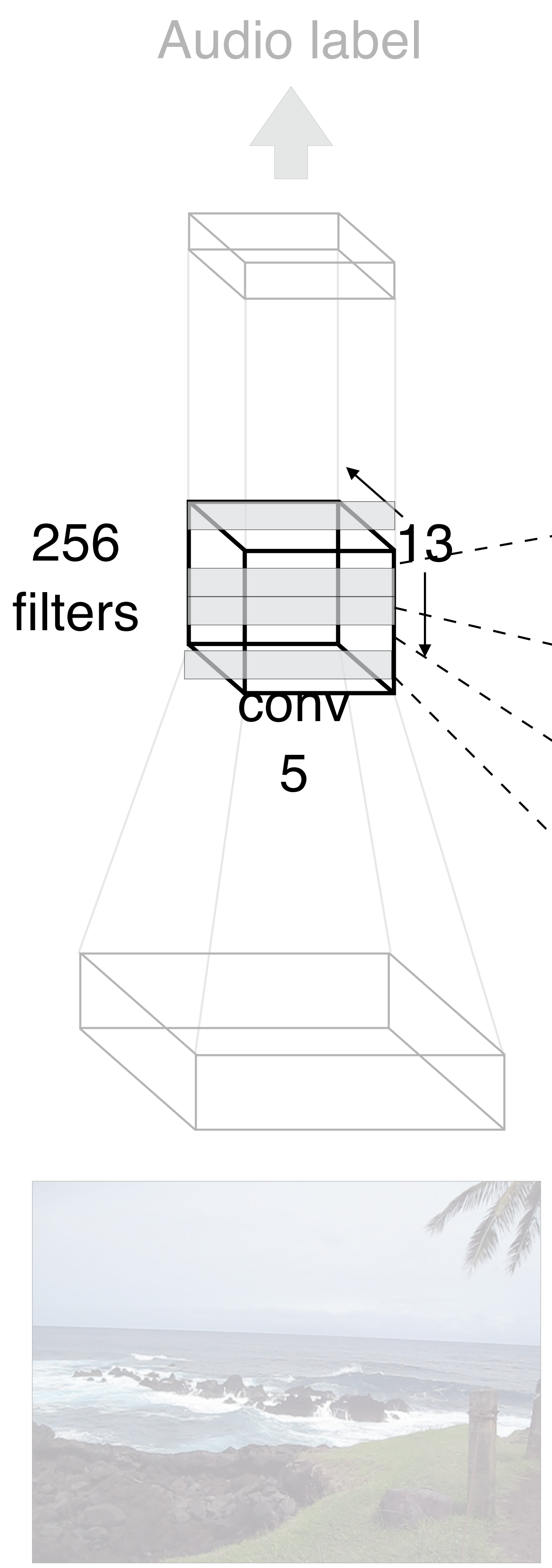
Unit visualizations

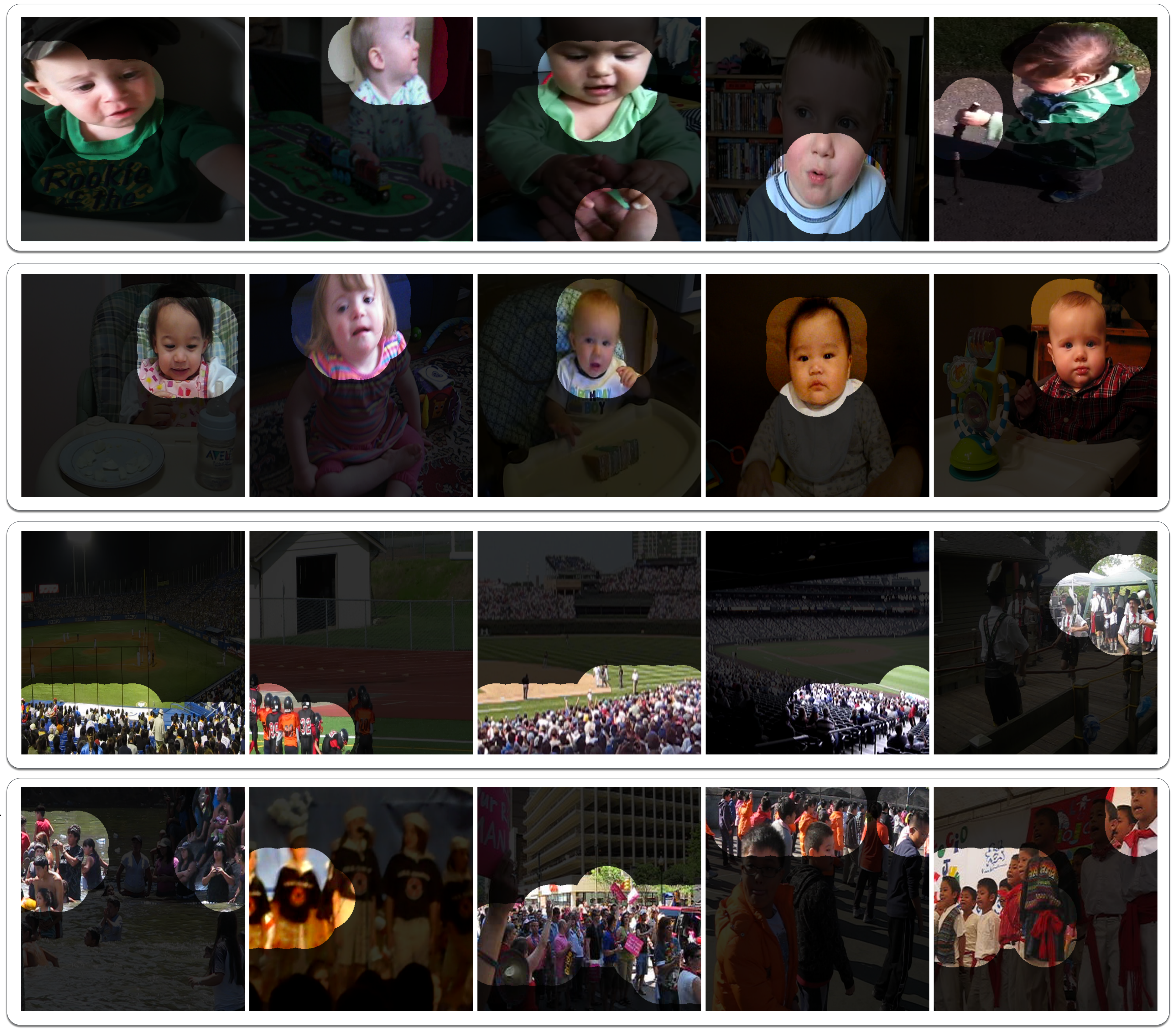
Top responses (unit #90 of 256)



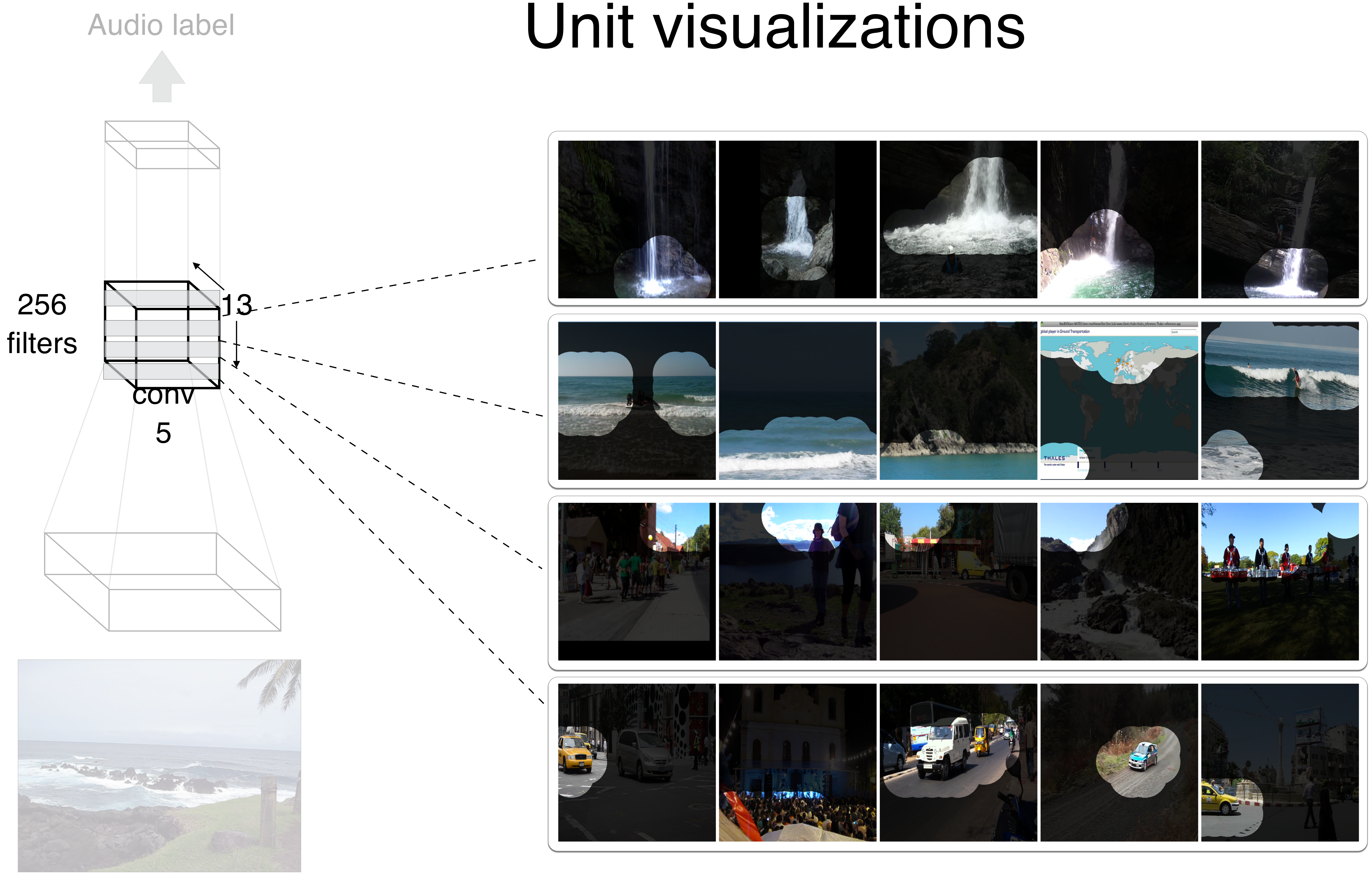


Unit visualizations



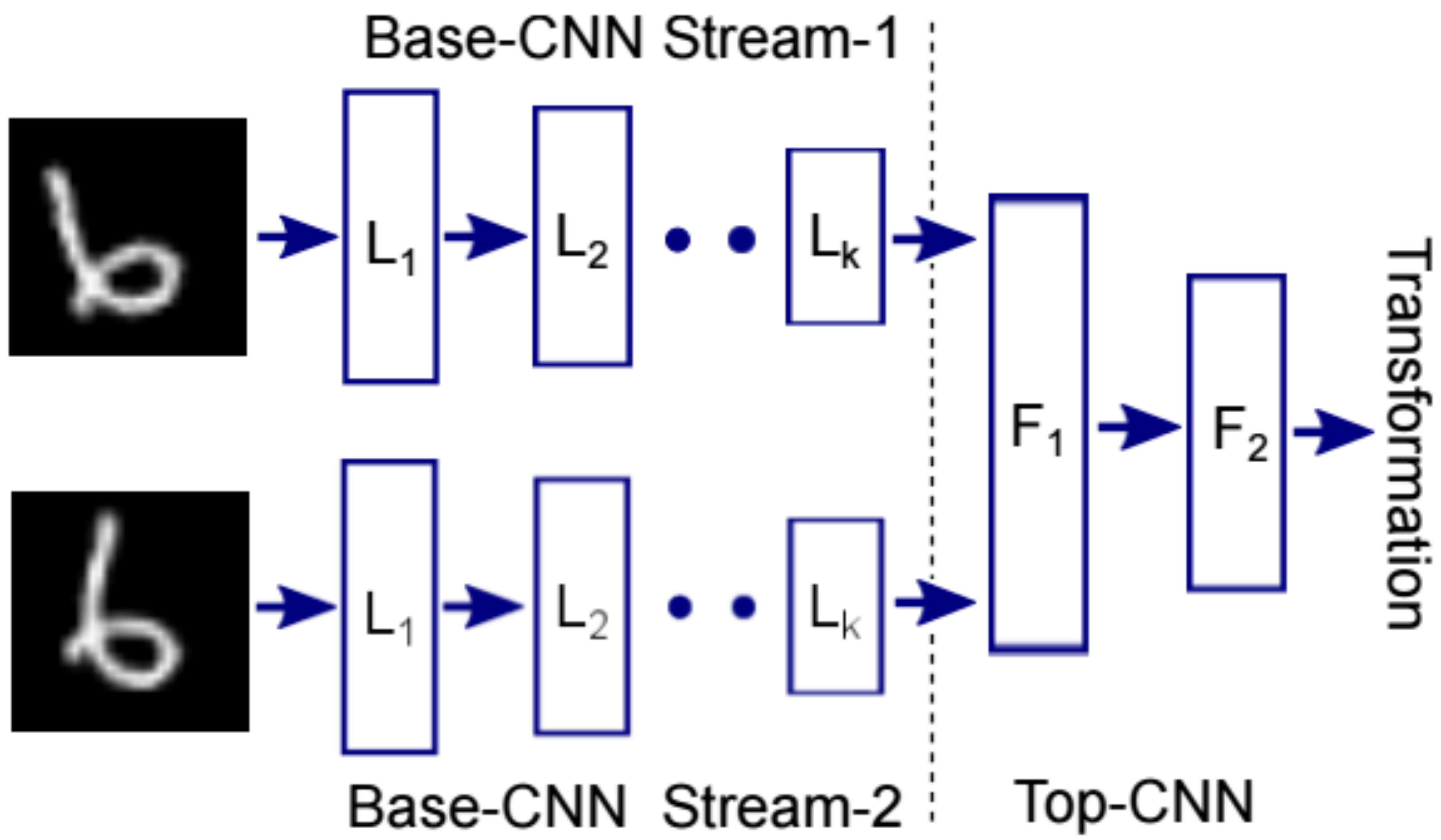


Unit visualizations



Cross-Modality

- Ego-motion
  - "We move in order to see and we see in order to move" - J.J Gibson
  - Ego-motion data is easy to collect
  - Siamese CNN to predict camera translation & Rotation along 3-axises. (Visual

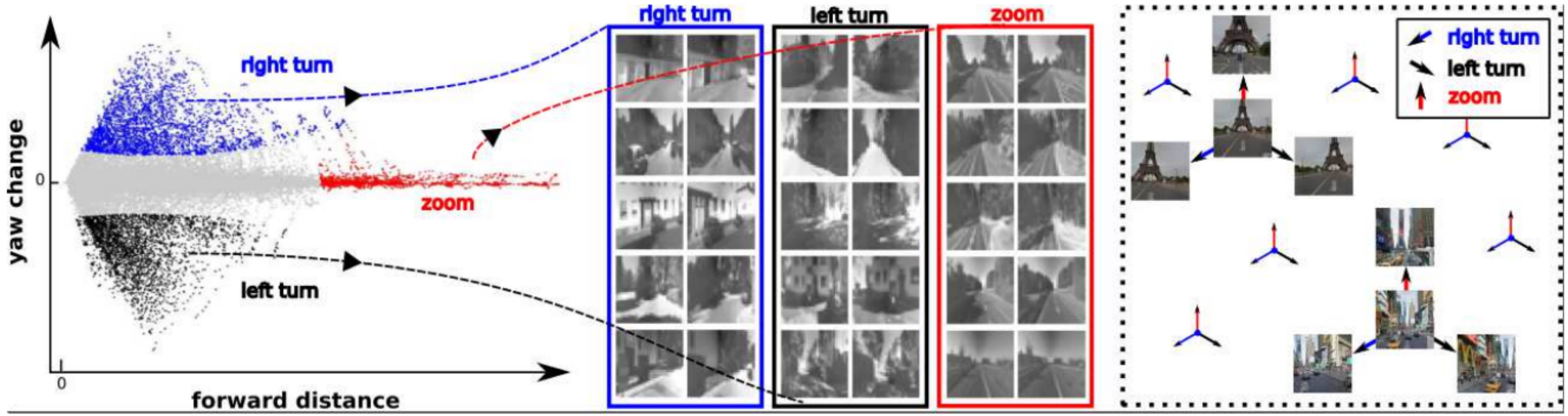


Agrawal, P., Carreira, J., & Malik, J. Learning to see by moving. In *ICCV 2015*

[A Survey to Self-Supervised Learning, Naiyan Wang]

Cross-Modality

- Ego-motion
  - Learning features that are equivariant to ego-motion



Jayaraman, D., & Grauman, K. Learning image representations tied to ego-motion. In *ICCV 2015*

[A Survey to Self-Supervised Learning, Naiyan Wang]

Cross-Modality

- Ego-motion
  - Siamese networks with contrastive loss
  - M_g is the transformation matrix specified by the external sensors

$$(\boldsymbol{\theta}^*, \mathcal{M}^*) = \arg\min_{\boldsymbol{\theta}, \mathcal{M}} \sum_{g,i,j} d_g \left(M_g \mathbf{z}_{\boldsymbol{\theta}}(\boldsymbol{x}_i), \mathbf{z}_{\boldsymbol{\theta}}(\boldsymbol{x}_j), p_{ij} \right),$$

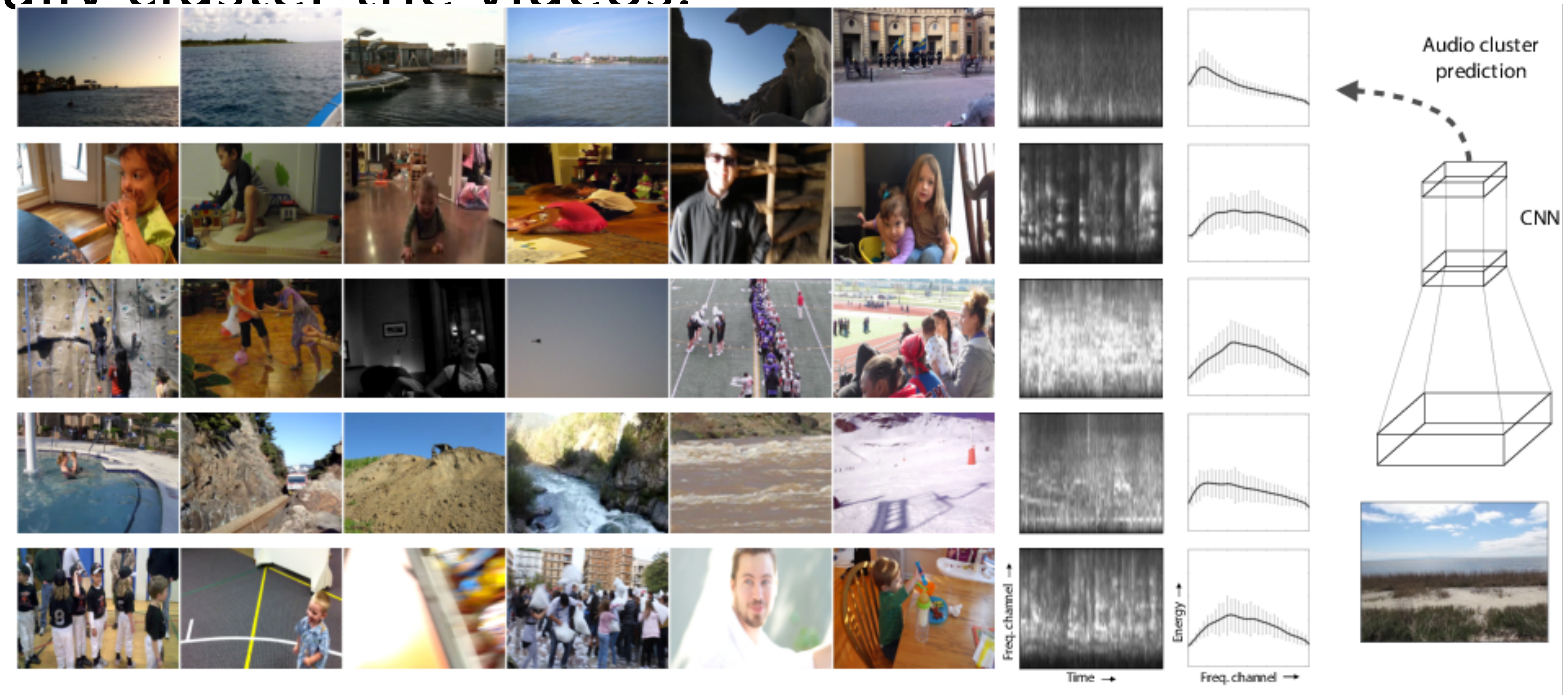
$$d_g(\boldsymbol{a}, \boldsymbol{b}, c) = \mathbb{1}(c = g) d(\boldsymbol{a}, \boldsymbol{b}) + \mathbb{1}(c \neq g) \max(\delta - d(\boldsymbol{a}, \boldsymbol{b}), 0),$$

Jayaraman, D., & Grauman, K. Learning image representations tied to ego-motion. In *ICCV 2015*

[A Survey to Self-Supervised Learning, Naiyan Wang]

Cross-Modality

- Acoustics -> RGB
  - Similar events should have similar sound.
  - Naturally cluster the videos.

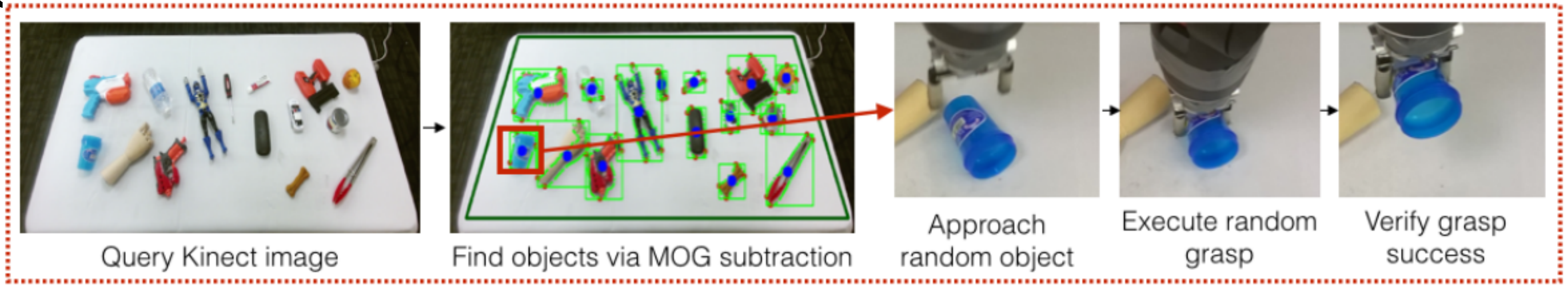


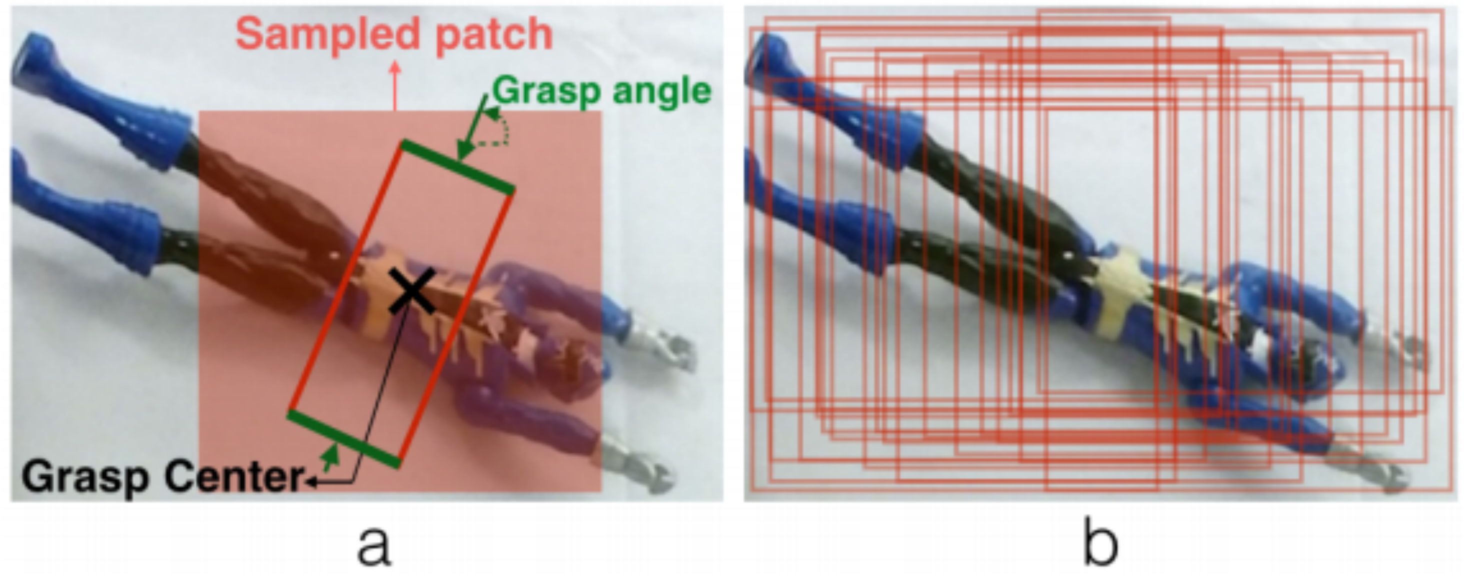
Owens, A., Wu, J., McDermott, J. H., Freeman, W. T., & Torralba, A. Ambient sound provides supervision for visual learning. In *ECCV 2016*

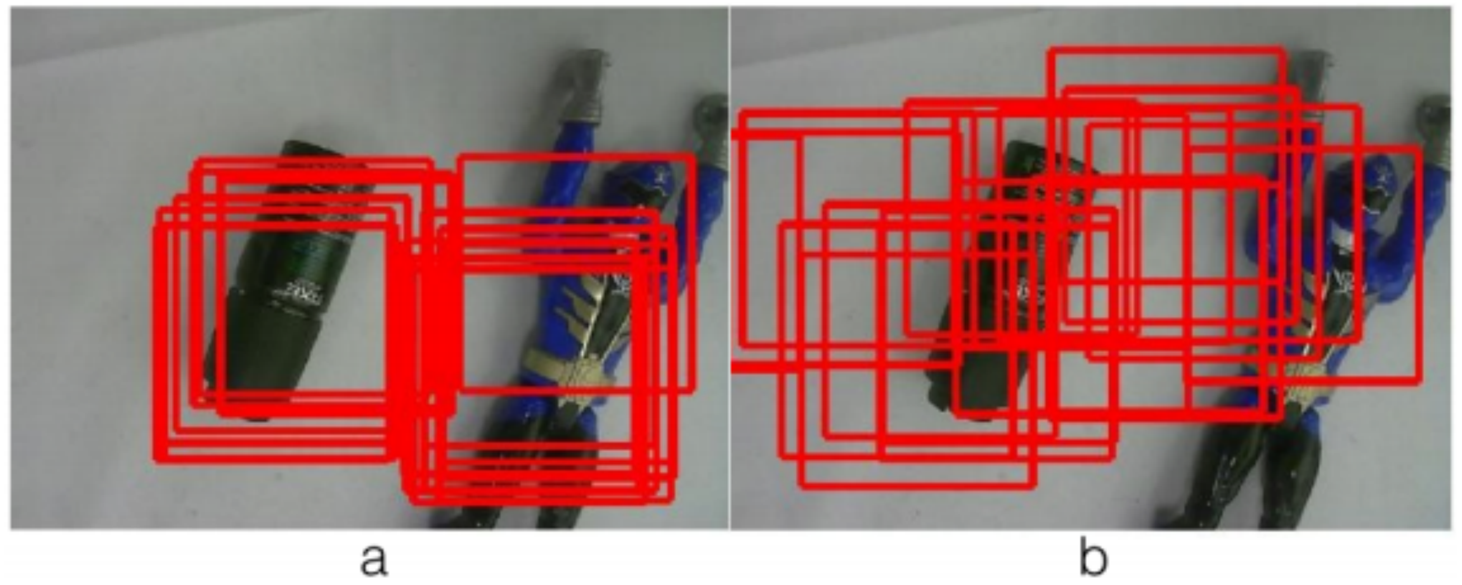
[A Survey to Self-Supervised Learning, Naiyan Wang]

Cross-Modality

- Features for grasping
  - Verify whether we could grasp the center of a patch at a given angle







Pinto, L., & Gupta, A. Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours. In *ICRA 2016*

[A Survey to Self-Supervised Learning, Naiyan Wang]

Evaluation

- Evaluate on general high-level vision tasks (classification, detection)
  - Be caution of different settings!

	Full train set						150 image set						
Method	All	>c1	>c2	>c3	>c4	>c5	All	>c1	>c2	>c3	>c4	>c5	#wins
Supervised													
Imagenet	56.5	57.0	57.1	57.1	55.6	52.5	17.7	19.1	19.7	20.3	20.9	19.6	NA
Sup. Masks (Ours)	51.7	51.8	52.7	52.2	52.0	47.5	13.6	13.8	15.5	17.6	18.1	15.1	NA
Unsupervised													
Jigsaw‡ [30]	49.0	50.0	48.9	47.7	45.8	37.1	5.9	8.7	8.8	10.1	9.9	7.9	NA
Kmeans [23]	42.8	42.2	40.3	37.1	32.4	26.0	4.1	4.9	5.0	4.5	4.2	4.0	0
Egomotion [2]	37.4	36.9	34.4	28.9	24.1	17.1	–	–	–	–	–	–	0
Inpainting [35]	39.1	36.4	34.1	29.4	24.8	13.4	–	–	–	–	–	–	0
Tracking-gray [46]	43.5	44.6	44.6	44.2	41.5	35.7	3.7	5.7	7.4	9.0	9.4	9.0	0
Sounds [33]	42.9	42.3	40.6	37.1	32.0	26.5	5.4	5.1	5.0	4.8	4.0	3.5	0
BiGAN [10]	44.9	44.6	44.7	42.4	38.4	29.4	4.9	6.1	7.3	7.6	7.1	4.6	0
Colorization [51]	44.5	44.9	44.7	44.4	42.6	38.0	6.1	7.9	8.6	10.6	10.7	9.9	0
Split-Brain Auto [52]	43.8	45.6	45.6	46.1	44.1	37.6	3.5	7.9	9.6	10.2	11.0	10.0	0
Context [8]	**49.9**	**48.8**	44.4	44.3	42.1	33.2	6.7	**10.2**	9.2	9.5	9.4	8.7	3
Context-videos† [8]	47.8	47.9	46.6	**47.2**	44.3	33.4	6.6	9.2	10.7	12.2	11.2	9.0	1
Motion Masks (Ours)	48.6	48.2	**48.3**	47.0	**45.8**	**40.3**	**10.2**	**10.2**	**11.7**	**12.5**	**13.3**	**11.0**	**9**

Pathak, D., Girshick, R., Dollár, P., Darrell, T., & Hariharan, B. Learning Features by Watching Objects Move. *In CVPR 2017*.

[A Survey to Self-Supervised Learning, Naiyan Wang]

Main issue with all these methods

- All these models rely on expert knowledge
- Need to define $y(x)$ for each new domain
- Not clear how to select a $y(x)$ that is a good target to learn all-purpose features

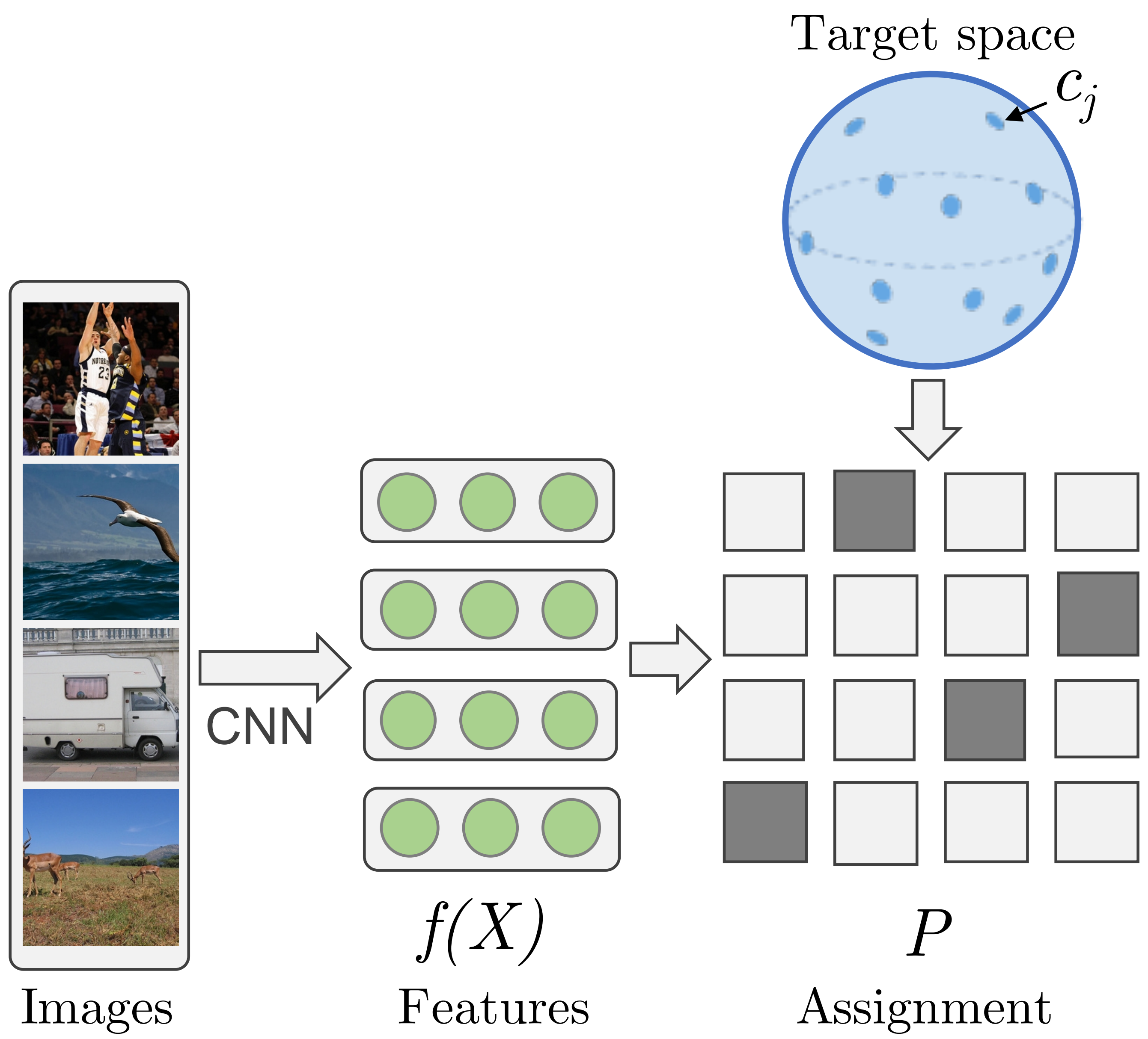
Unsupervised Learning by Predicting Noise

Piotr Bojanowski, Armand Joulin

ICML 2017

Unsupervised Learning by Predicting Noise

[Bojanowski & Joulin, ICML 2017]



- Inspired by Dosovitskiy et al.
- Learn mapping from images to a sphere
- Fix targets on sphere
- Simultaneously:
  - Learn the mapping
  - Optimize the assignment between images and targets

Deep Discriminative Clustering

- We are given a set of n images

$$\{x_1, \ldots, x_n\}$$

- We want to learn a visual features f without using labels

$$\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} \min_{y_i} \ell(f_\theta(x_i), y_i)$$

$$\min_{\theta} \min_{Y} \frac{1}{2n} \|f_\theta(X) - Y\|_F^2$$

- We use the L2 loss

[Bojanowski & Joulin, ICML 2017]

Label Collapse Problem

- Optimization over Y would lead to a collapse
- Repulsive costs are tricky to use
- Can impose constraints on Y but hard to optimize

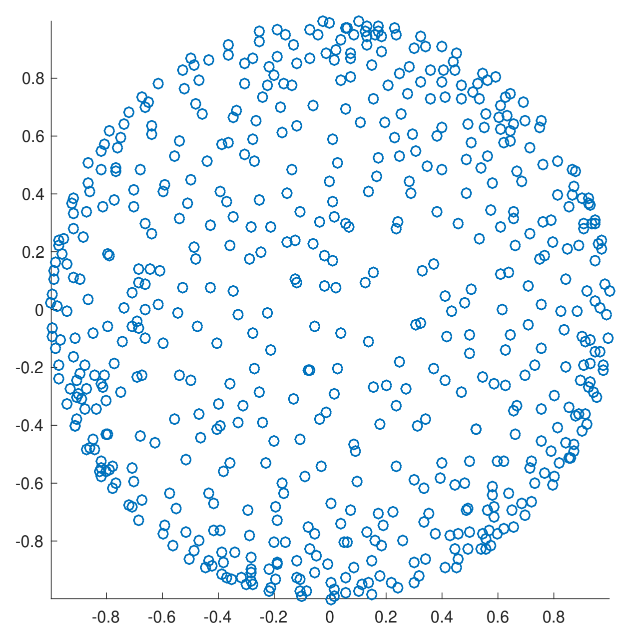
[Bojanowski & Joulin, ICML 2017]

Fixing the Target Representation

- Instead, we fix the target representation
- Allow a reassignment between targets and images

$$Y = PC \qquad \mathcal{P} = \{P \in \{0,1\}^{n \times k} \mid P\mathbf{1} = \mathbf{1}, P^\top \mathbf{1} = \mathbf{1}\}$$

- Targets C are **uniformly sampled on the sphere**



- Final objective function

$$\min_\theta \min_{P \in \mathcal{P}} \frac{1}{2n} \|f_\theta(X) - PC\|_F^2$$

[Bojanowski & Joulin, ICML 2017]

Optimization

- We minimize our cost function in an on-line fashion
- We use the following algorithm:

Require: T batches of images, $\lambda_0 > 0$
  for $t = \{1, \dots, T\}$ **do**
    Obtain batch b and representations r
    Compute $f_\theta(X_b)$
    Compute P^* by minimizing w.r.t. P
    Compute $\nabla_\theta L(\theta)$ using P^*
    Update $\theta \leftarrow \theta - \lambda_t \nabla_\theta L(\theta)$
  end for

[Bojanowski & Joulin, ICML 2017]

Optimizing the Permutation Matrix

- At theta fixed, the permutation is obtained by solving

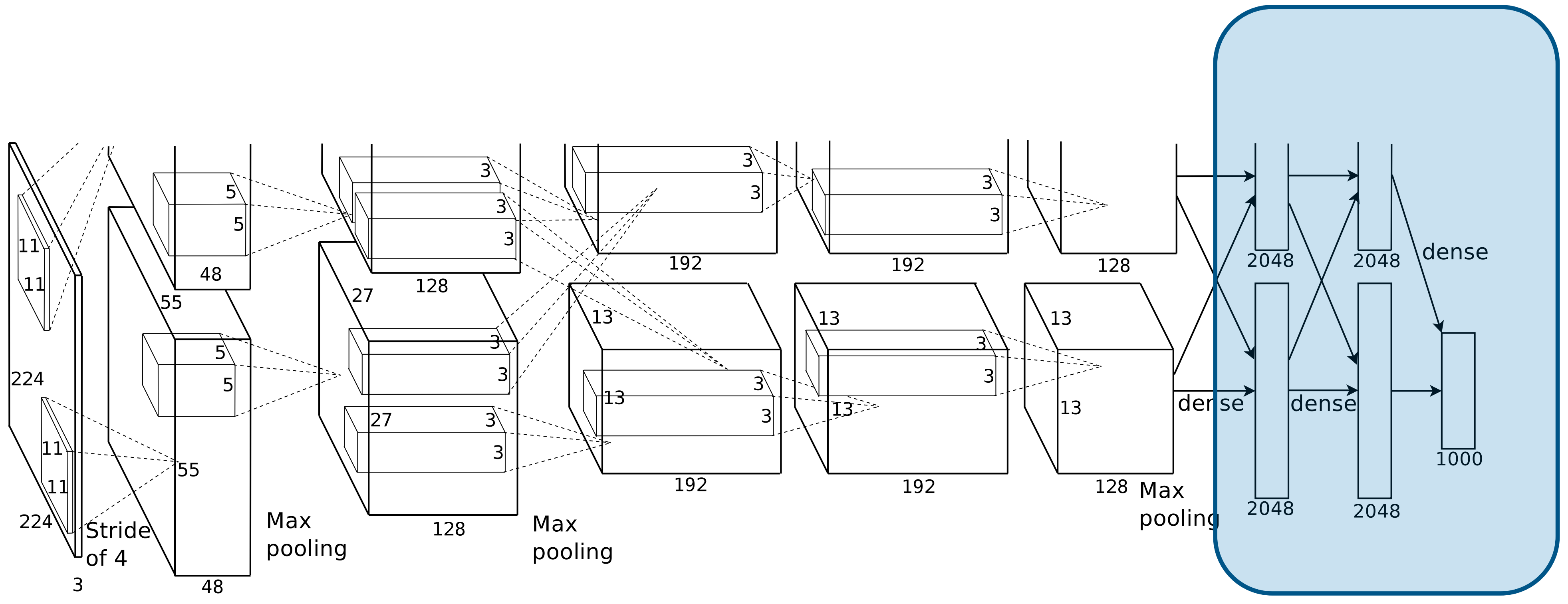
$$\max_{P \in \mathcal{P}} \ \mathrm{Tr}\left(PCf_\theta(X)^\top\right).$$

- Which is a linear program on the set of permutation matrices
- We can use the Hungarian algorithm

$$\mathcal{O}(nb^2)$$

[Bojanowski & Joulin, ICML 2017]

Experimental Setup



- AlexNet architecture
- Learn unsupervised features on ImageNet training set
- Retrain a classifier on top for a target transfer task, i.e. PASCAL VOC Classification / Detection

[Bojanowski & Joulin, ICML 2017]

Baselines

- Self supervised models
  - Wang & Gupta – Temporal coherence in videos
  - Doersch et al. – Predict context patches
  - Zhang et al. – Predict color
  - Norouzi & Favaro – Solve jigsaw puzzles

- Unsupervised model
  - GAN
  - Auto-encoder
  - BI-GAN (Donahue et al.)

[Bojanowski & Joulin, ICML 2017]

Pascal VOC - results

	Classification		Detection
Trained layers	fc6-8	all	all
ImageNet labels	78.9	79.9	56.8
Agrawal et al.	31.0	54.2	43.9
Pathak et al.	34.6	56.5	44.5
Wang & Gupta	55.6	63.1	47.4
Doersch et al.	55.1	65.3	51.1
Zhang et al.	61.5	65.6	46.9
Autoencoder	16.0	53.8	41.9
GAN	40.5	56.4	-
BiGAN	52.3	60.1	46.9
NAT	56.7	65.3	49.4

- compare favorably to SOTA
- Poor performance of AE / GAN

[Bojanowski & Joulin, ICML 2017]

Nearest Neighbor Queries

[Bojanowski & Joulin, ICML 2017]

Bojanowski & Joulin Summary

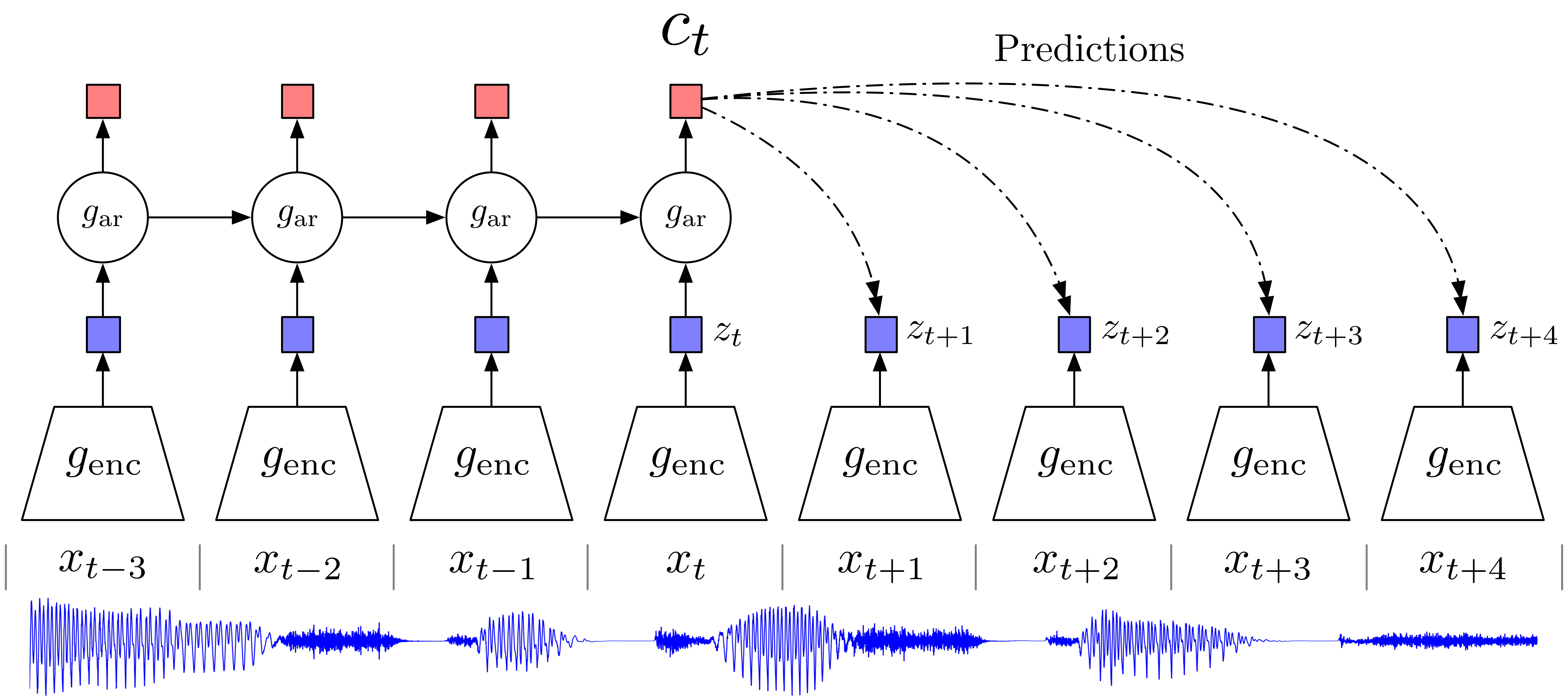
- Simple unsupervised approach
- No domain expert knowledge
- Scales to **very large** datasets
- Close to supervised pipeline
- SOTA performance (at the time) amongst unsupervised methods

Contrastive Predictive Coding

Aaron van den Oord, Yazhe Li, Oriol Vinyals
Google DeepMind

https://arxiv.org/pdf/1807.03748.pdf

Model Overview



CPC Principle

- Encode the target x (future) and context c (present) into a compact distributed vector representations (via non-linear learned mappings) in a way that maximally preserves the mutual information of the original signals x and c defined as

$$I(x; c) = \sum_{x,c} p(x, c) \log \frac{p(x|c)}{p(x)}.$$

- By maximizing the mutual information between the encoded representations (which is bounded by the MI between the input signals), we extract the underlying latent variables the inputs have in common.

CPC Model

- Do NOT predict future observations x_{t+k} directly with a generative model $p_k(x_{t+k}|c_t)$
- Instead we model a **density ratio** which preserves the mutual information between x_{t+k} and c_t (prev eqn):

$$f_k(x_{t+k}, c_t) \propto \frac{p(x_{t+k}|c_t)}{p(x_{t+k})}$$

Where: $$f_k(x_{t+k}, c_t) = \exp\left(z_{t+k}^T W_k c_t\right),$$

Noise Contrastive Estimation Loss

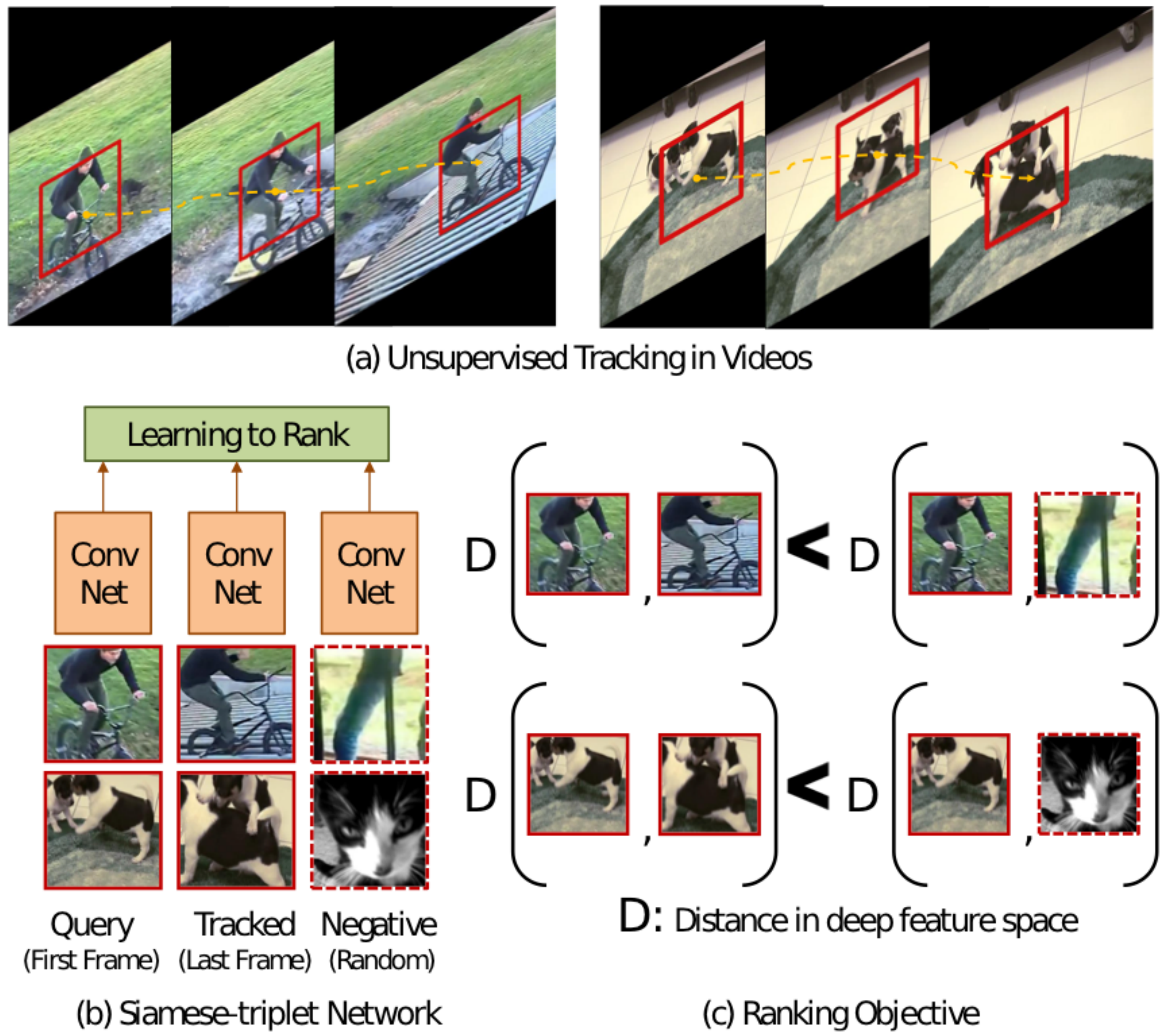
- Given a set $X = \{x_1, \ldots x_N\}$ of N random samples containing one positive sample from $p(x_{t+k} | c_t)$ and $N - 1$ negative samples from the 'proposal' distribution $p(x_{t+k})$, we optimize

$$\mathcal{L}_\mathrm{N} = -\mathbb{E}_{X}\left[\log \frac{f_k(x_{t+k}, c_t)}{\sum_{x_j \in X} f_k(x_j, c_t)}\right]$$

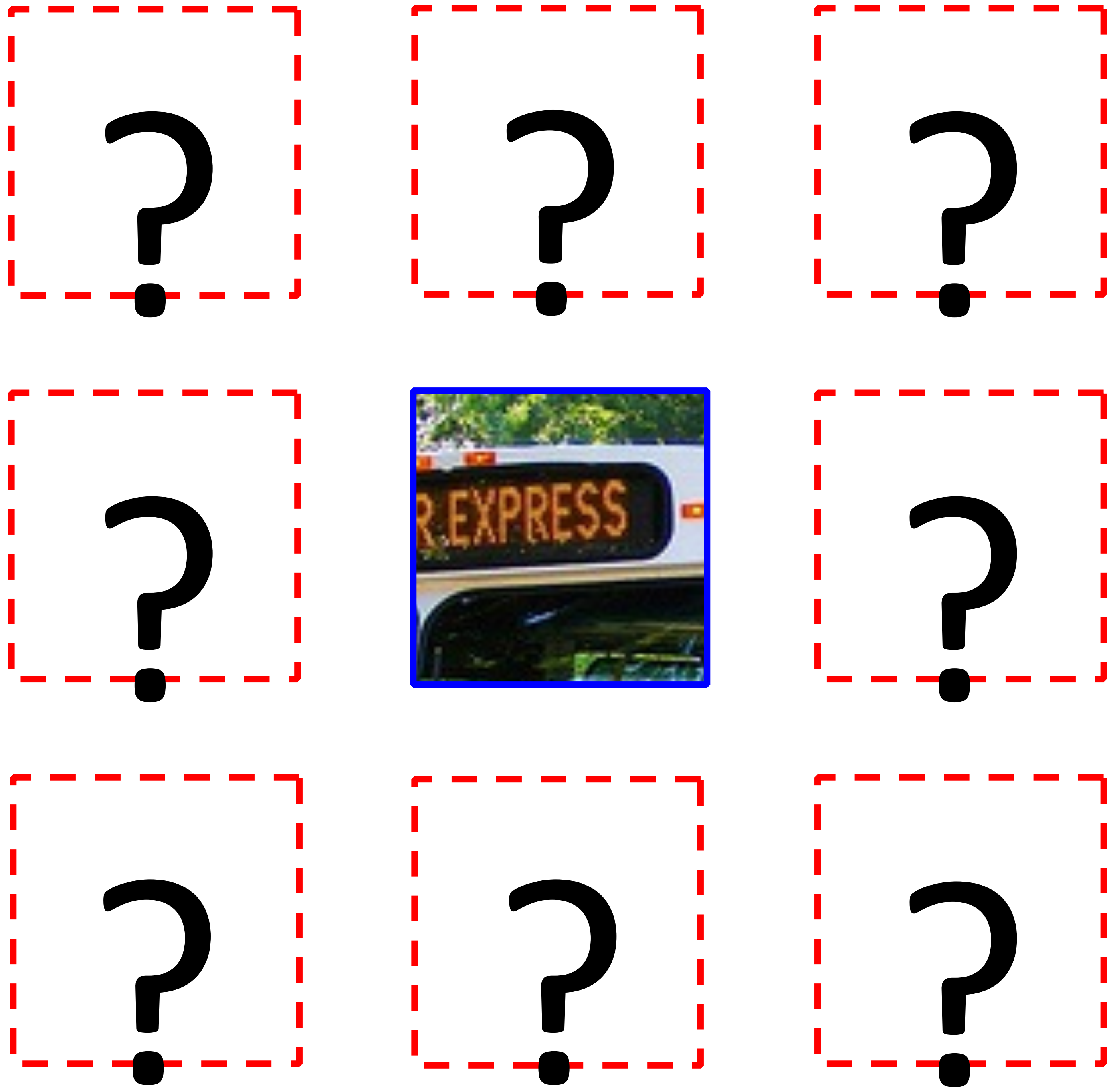
  Optimizing this loss will result in $f_k(x_{t+k}, c_t)$ estimating the density ratio in prev slide.
- Not always clear in expts where the N random samples come from
  - E.g. same/different sequence? Narrow window?

C.F. Vision SSL approaches

[Wang & Gupta 2015]



Unsupervised Visual Representation Learning by Context Prediction [Doersch et al. ICCV 2015]



CPC applied to Audio

Method	ACC
Phone classification	
Random initialization	27.6
MFCC features	39.7
CPC	64.6
Supervised	74.6
Speaker classification	
Random initialization	1.87
MFCC features	17.6
CPC	97.4
Supervised	98.5

Table 1: LibriSpeech phone and speaker classification results. For phone classification there are 41 possible classes and for speaker classification 251. All models used the same architecture and the same audio input sizes.

Method	ACC
#steps predicted	
2 steps	28.5
4 steps	57.6
8 steps	63.6
12 steps	64.6
16 steps	63.8
Negative samples from	
Mixed speaker	64.6
Same speaker	65.5
Mixed speaker (excl.)	57.3
Same speaker (excl.)	64.6
Current sequence only	65.2

Table 2: LibriSpeech phone classification ablation experiments. More details can be found in Section 3.1.

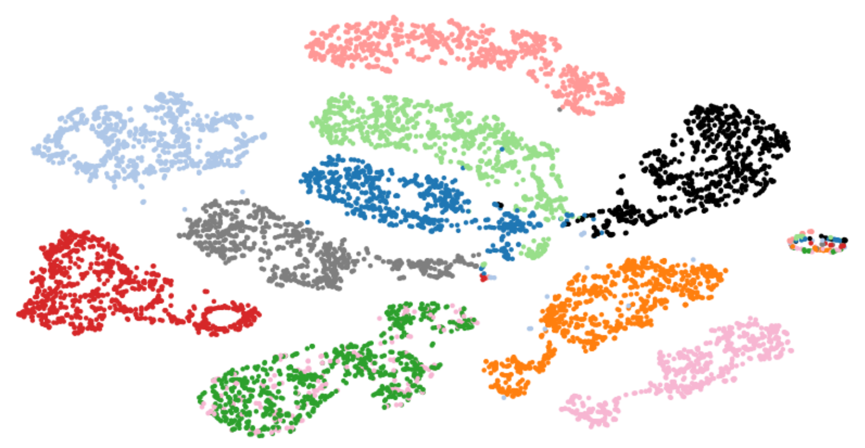


Figure 2: t-SNE visualization of audio (speech) representations for a subset of 10 speakers (out of 251). Every color represents a different speaker.

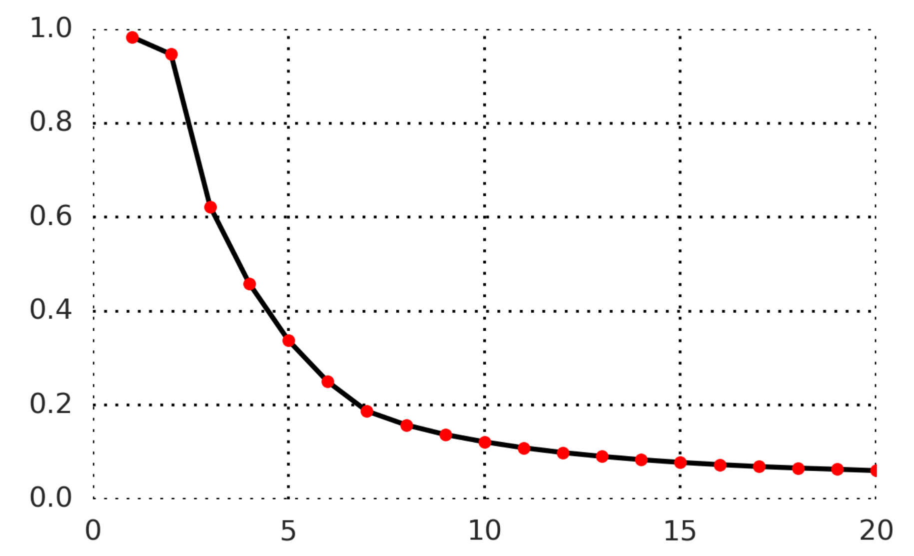
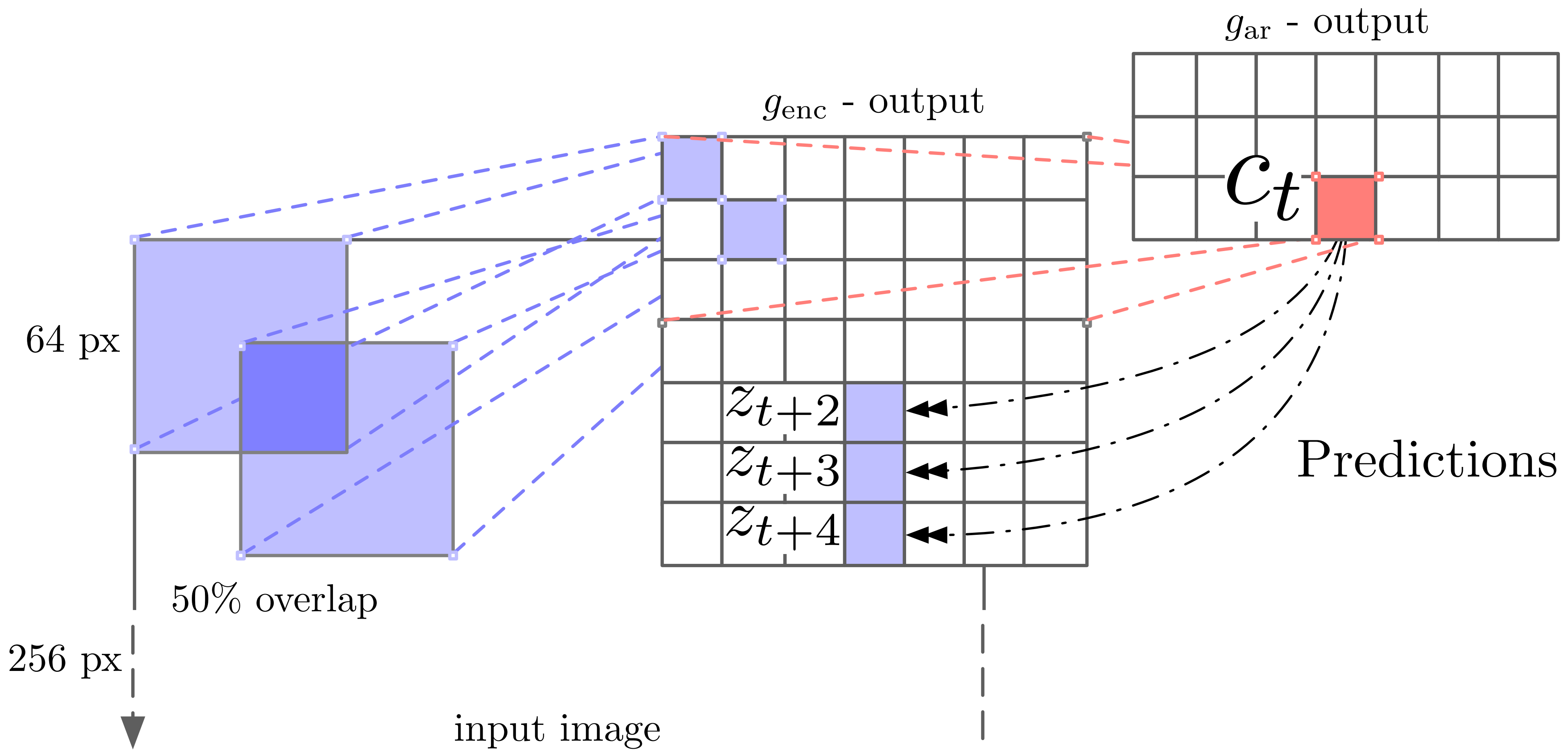


Figure 3: Average accuracy of predicting the positive sample in the contrastive loss for 1 to 20 latent steps in the future of a speech waveform. The model predicts up to 200ms in the future as every step consists of 10ms of audio.

CPC applied to images



Method	Top-1 ACC
Using AlexNet conv5	
Video [27]	29.8
Relative Position [11]	30.4
BiGan [34]	34.8
Colorization [10]	35.2
Jigsaw [28] *	38.1
Using ResNet-V2	
Motion Segmentation [35]	27.6
Exemplar [35]	31.5
Relative Position [35]	36.2
Colorization [35]	39.6
CPC	**48.7**

Table 3: ImageNet top-1 unsupervised classification results. *Jigsaw is not directly comparable to the other AlexNet results because of architectural differences.

Google Research

SimCLR: A Simple Framework for Contrastive Learning of Visual Representations

Ting Chen
Simon
Kornblith
Mohammad Norouzi
Geoffrey Hinton

Google Research, Brain Team

The proposed SimCLR framework

A simple idea: maximizing the agreement of representations under data transformation, using a contrastive loss in the latent/feature space.

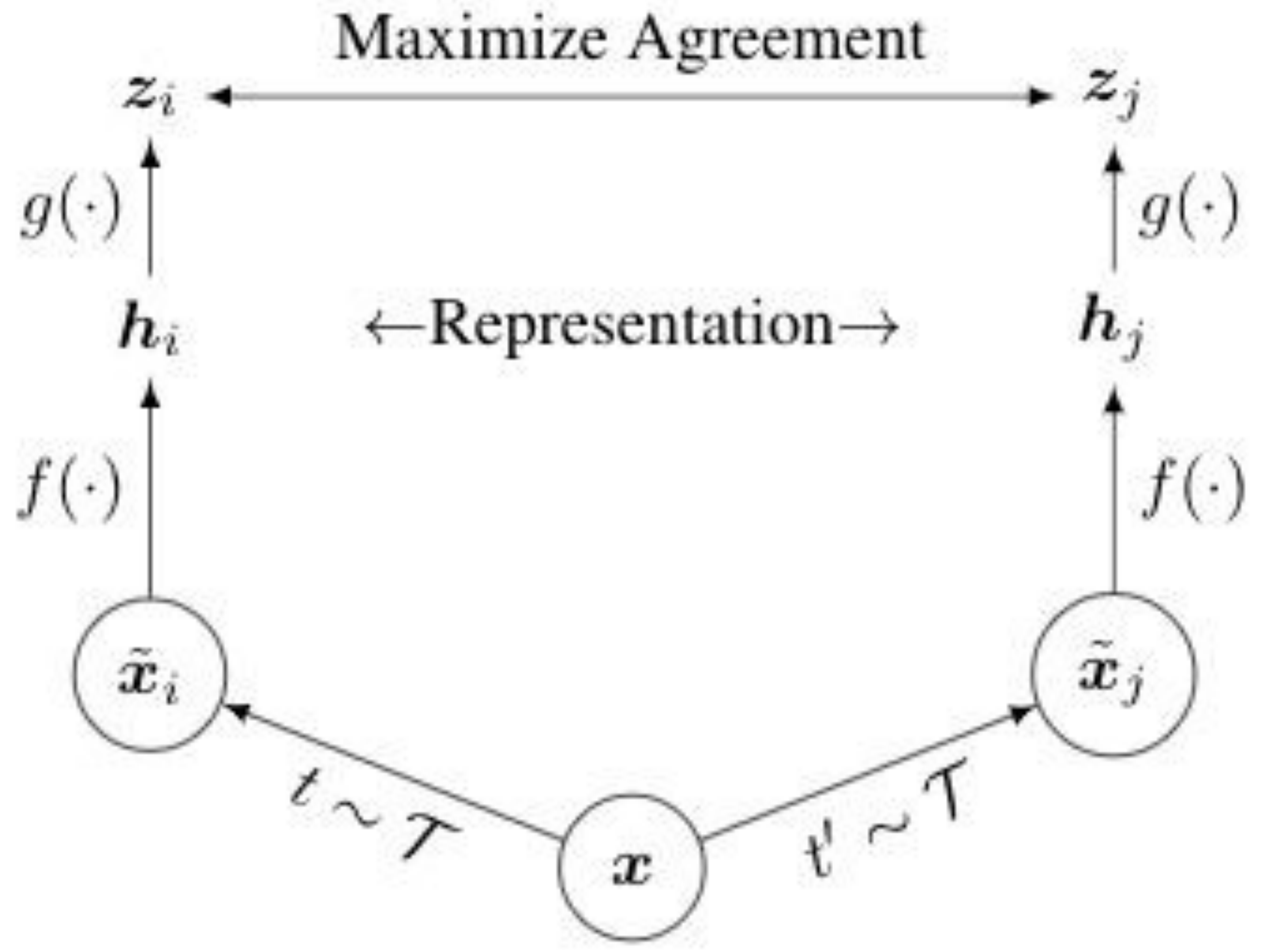
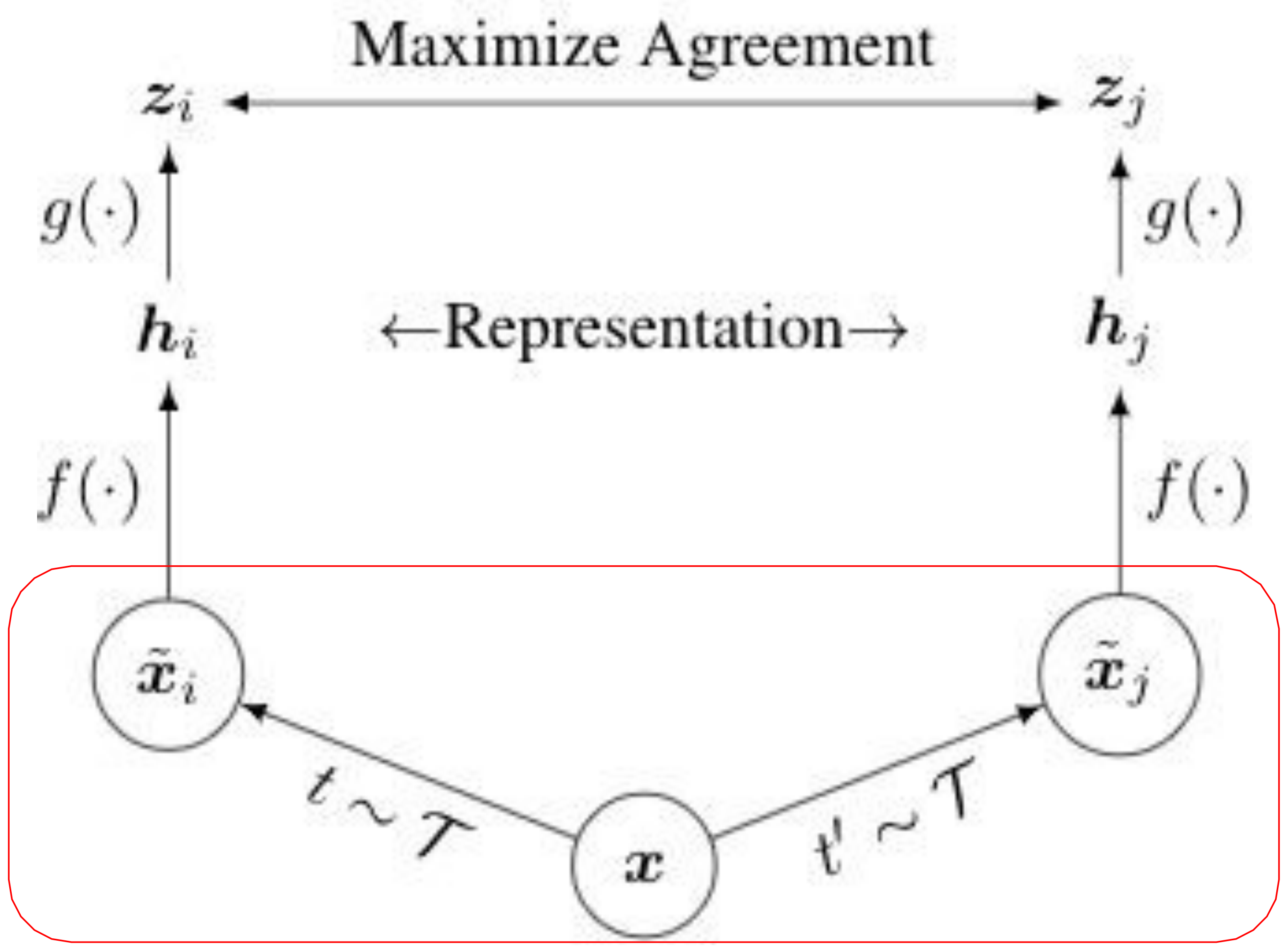


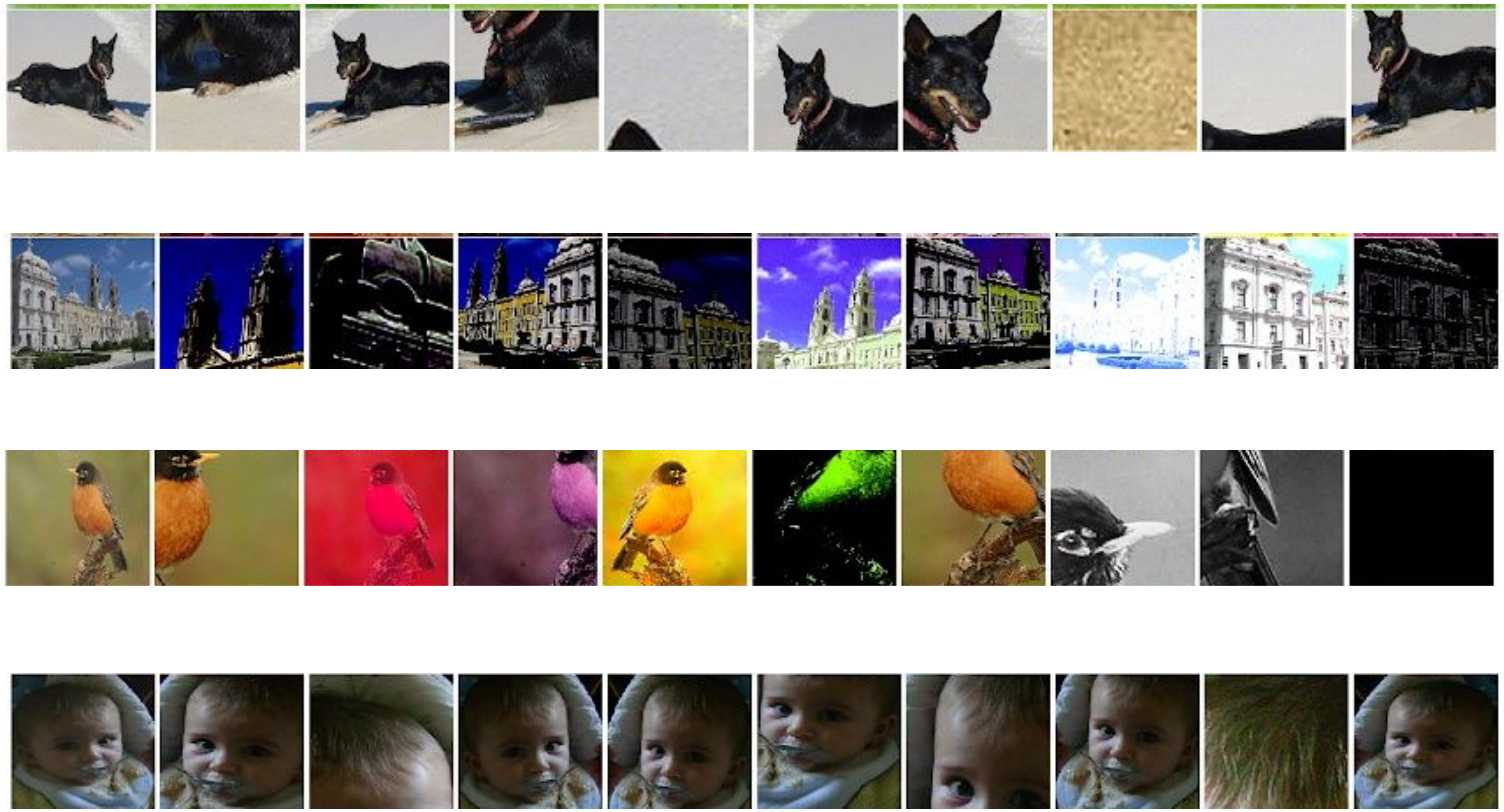
Figure 2. A framework for contrastive representation learning. Two separate stochastic data augmentations $t, t' \sim \mathcal{T}$ are applied to each example to obtain two correlated views. A base encoder network $f(\cdot)$ with a projection head $g(\cdot)$ is trained to maximize agreement in *latent representations* via a contrastive loss.

The proposed SimCLR framework

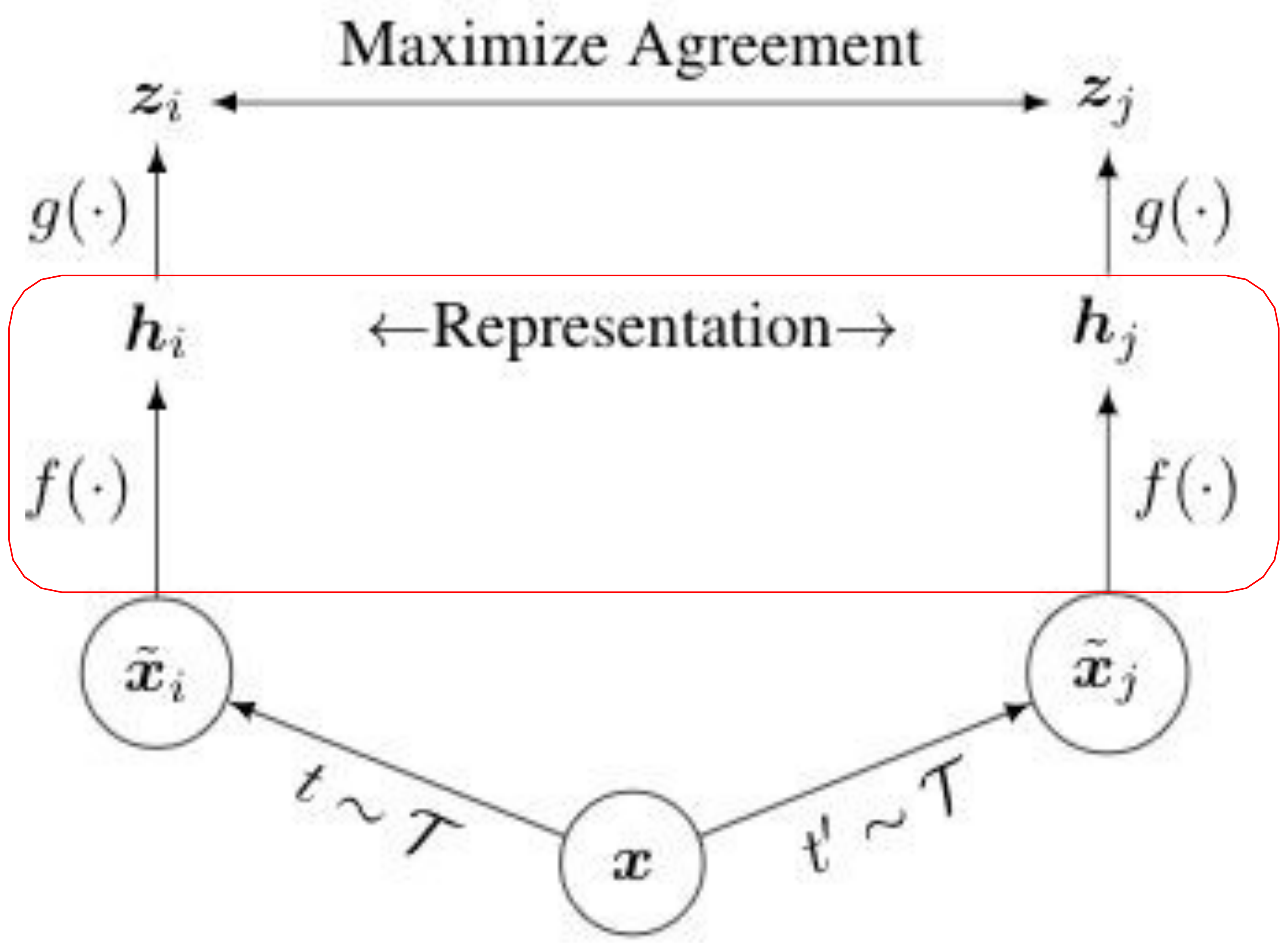


We use random crop and color distortion for augmentation.

Examples of augmentation applied to the left most images:

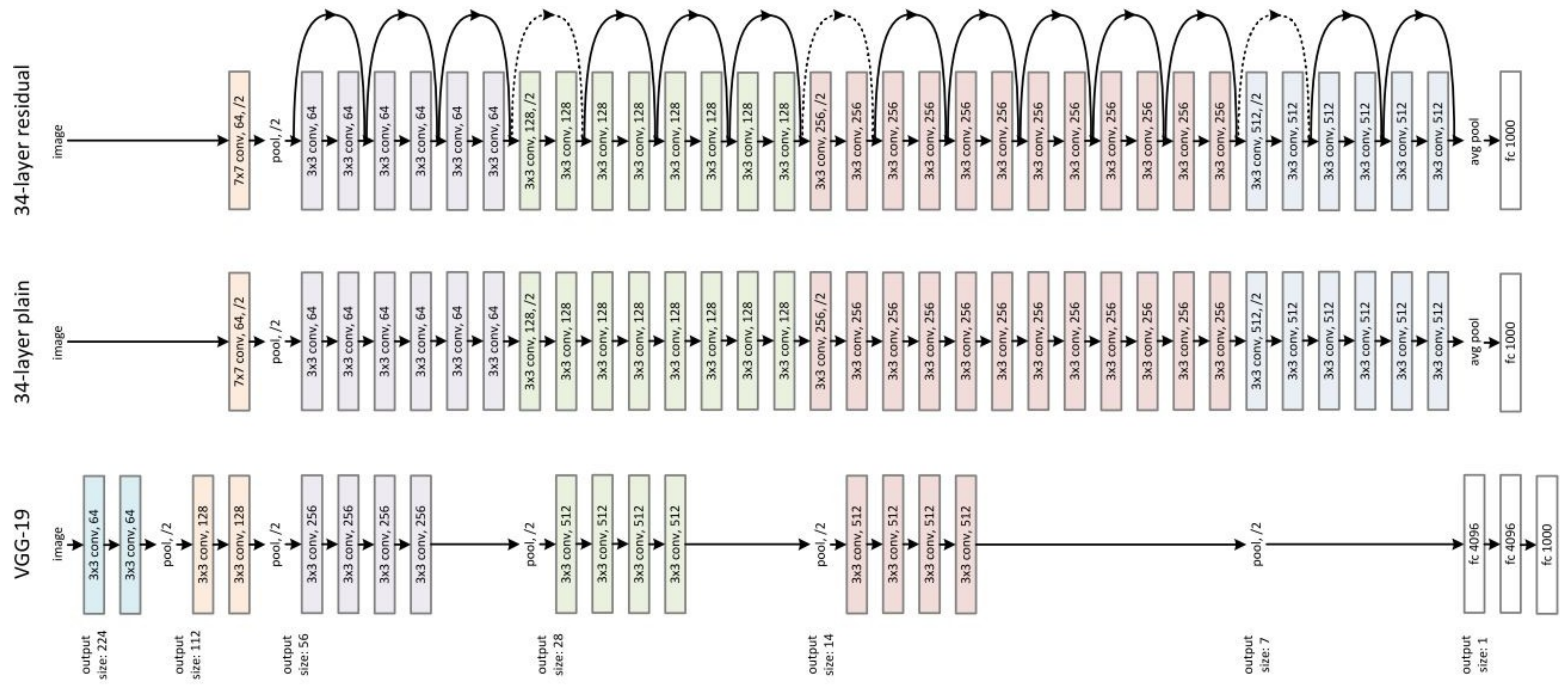


The proposed SimCLR framework

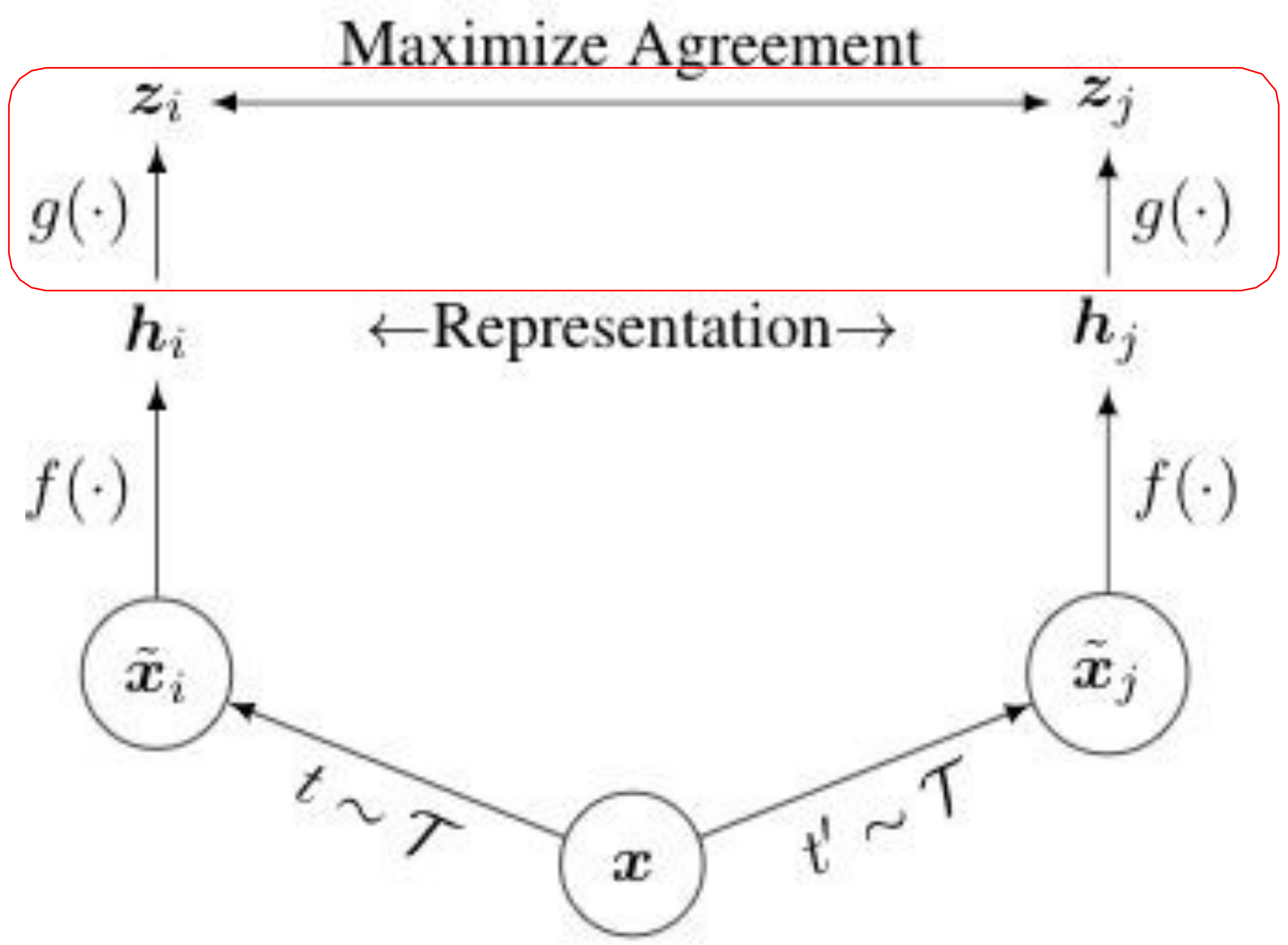


f(x) is the base network that computes internal representation.

We use (unconstrained) ResNet in this work. However, it can be other networks.

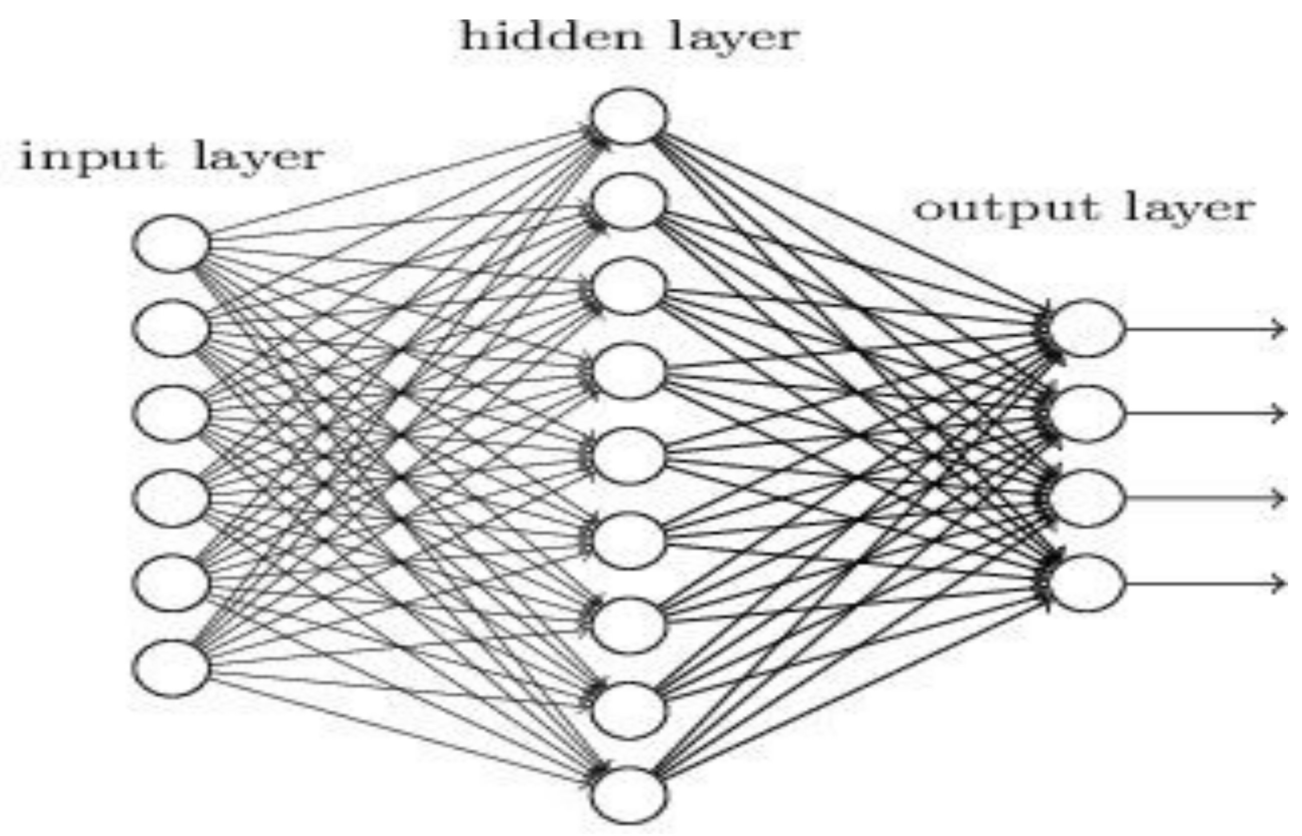


The proposed SimCLR framework



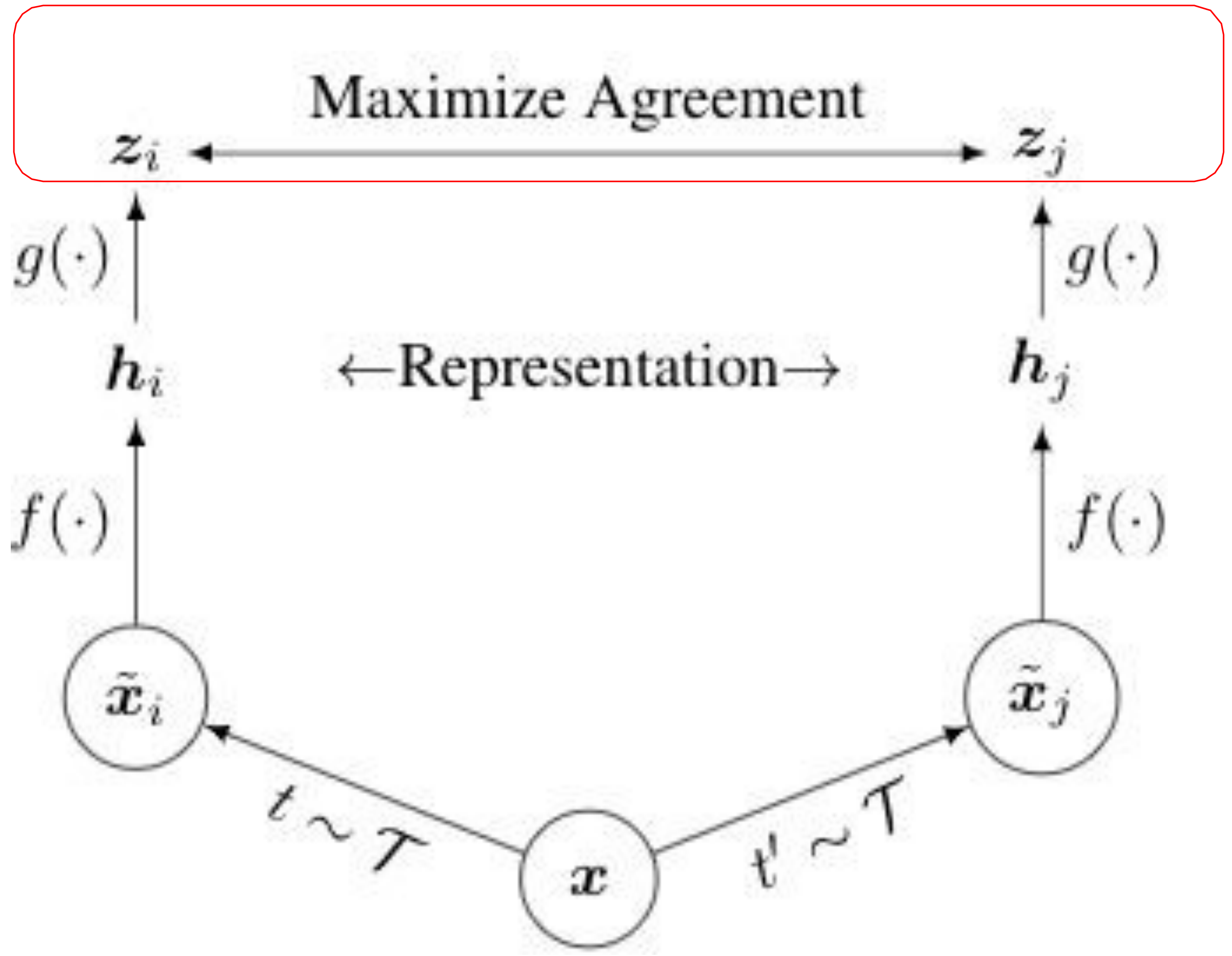
g(h) is a projection network that project representation to a latent space.

We use a 2-layer non-linear MLP (fully connected net).



Google Research

The proposed SimCLR framework



Maximize agreement using a contrastive task:

Given {x_k} where two different examples x_i and x_j are a positive pair, identify x_j in {x_k}_{k!=i} for x_i.

Original image crop 1 crop 2 contrastive image

Loss function:

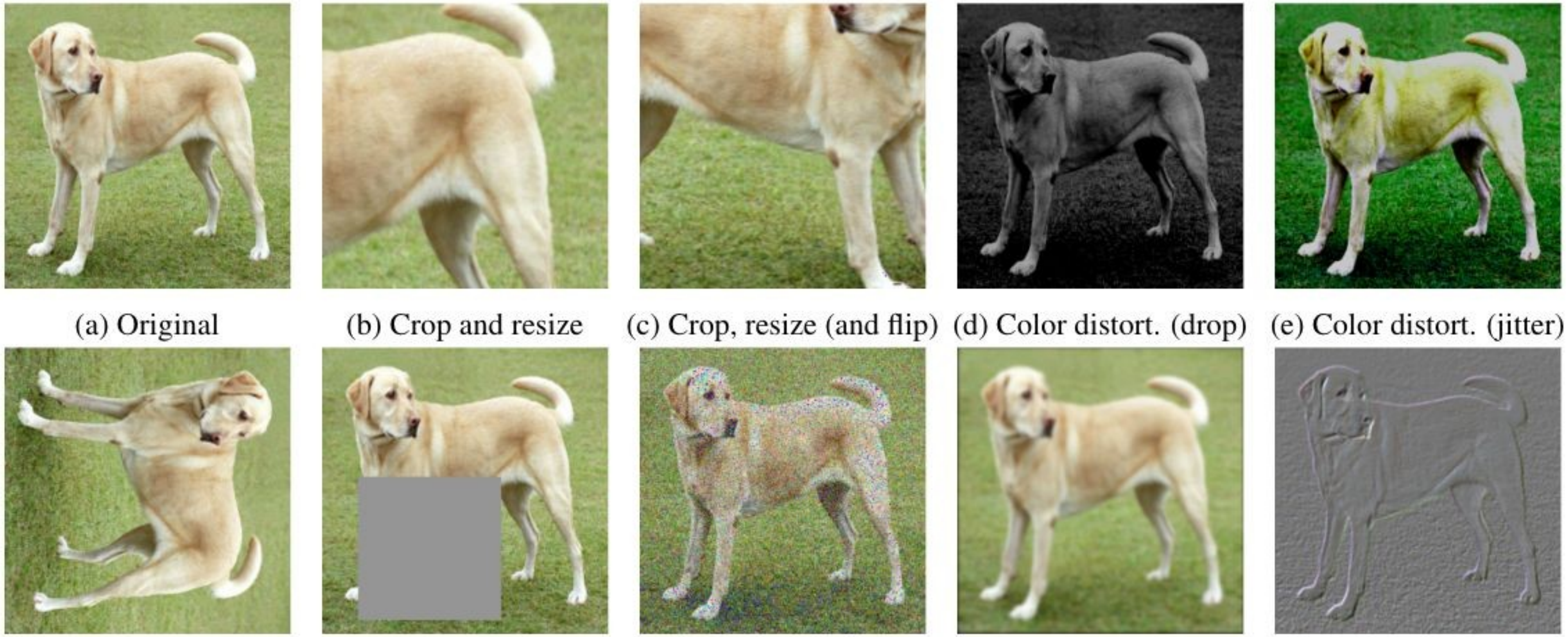
$$\text{Let } \mathrm{sim}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{u}^\top \boldsymbol{v} / \|\boldsymbol{u}\| \|\boldsymbol{v}\|$$

$$\ell_{i,j} = -\log \frac{\exp(\mathrm{sim}(\boldsymbol{z}_i, \boldsymbol{z}_j)/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\mathrm{sim}(\boldsymbol{z}_i, \boldsymbol{z}_k)/\tau)}$$

Data Augmentation for Contrastive Representation Learning

We study a set of transformations...

Systematically study a set of augmentation



(a) Original (b) Crop and resize (c) Crop, resize (and flip) (d) Color distort. (drop) (e) Color distort. (jitter)

(f) Rotate {90°, 180°, 270°} (g) Cutout (h) Gaussian noise (i) Gaussian blur (j) Sobel filtering

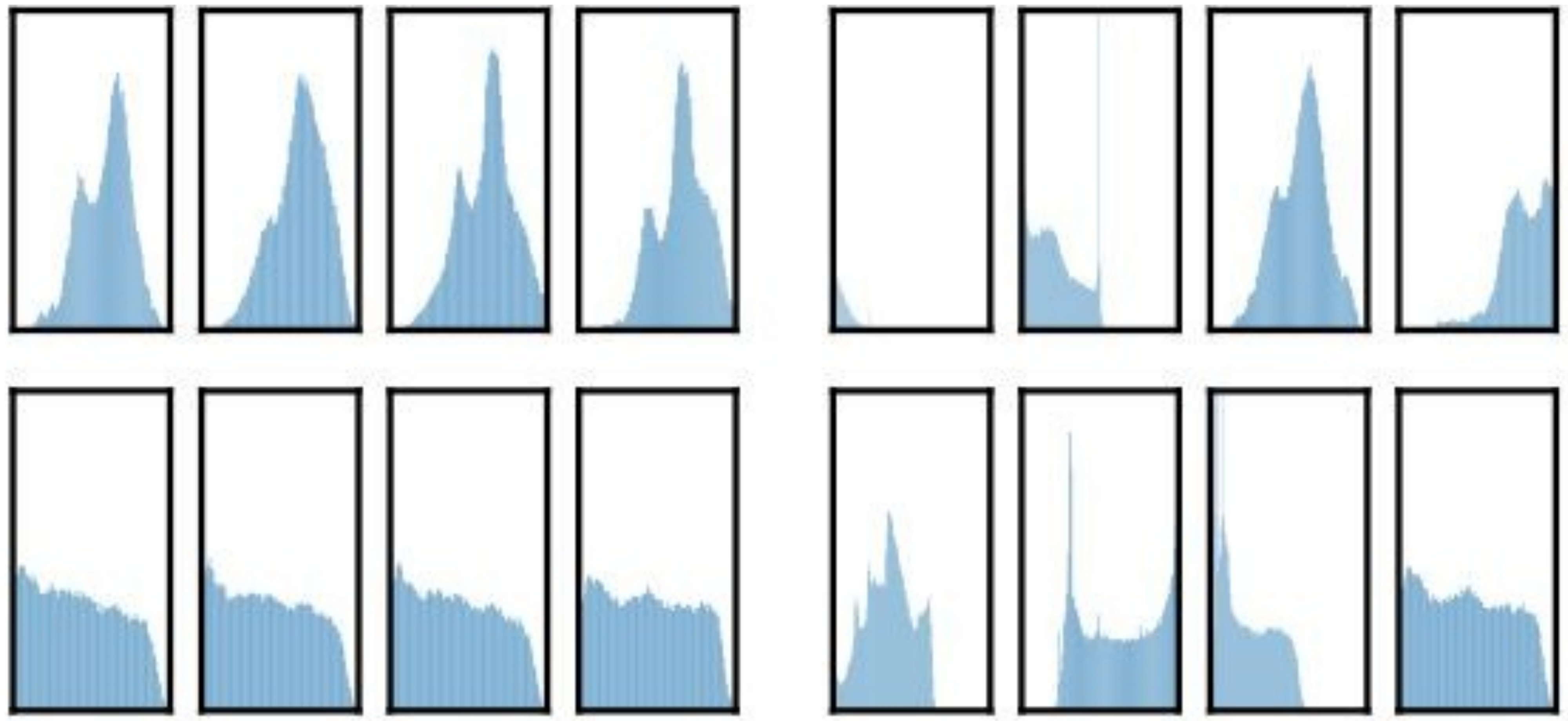
* Note that we only test these for ablation, the augmentation policy used to train our models only involves random crop (with flip and resize) + color distortion + Gaussian blur.

Google Research

Composition of augmentations are crucial

Composition of crop and color stands out!

Figure 5. Linear evaluation (ImageNet top-1 accuracy) under individual or composition of data augmentations, applied only to one branch. For all columns but the last, diagonal entries correspond to single transformation, and off-diagonals correspond to composition of two transformations (applied sequentially). The last column reflects the average over the row.



(a) Without color distortion. (b) With color distortion.

Figure 6. Histograms of pixel intensities (over all channels) for different crops of two different images (i.e. two rows). The image for the first row is from Figure 4. All axes have the same range.

Encoder and Projection Head

Google Research

A nonlinear projection head improves the representation quality of the layer before it

)

We compare three projection head g(.) (after average pooling of ResNet):

- Identity mapping
- Linear projection
- Nonlinear projection with one additional hidden layer (and ReLU activatio

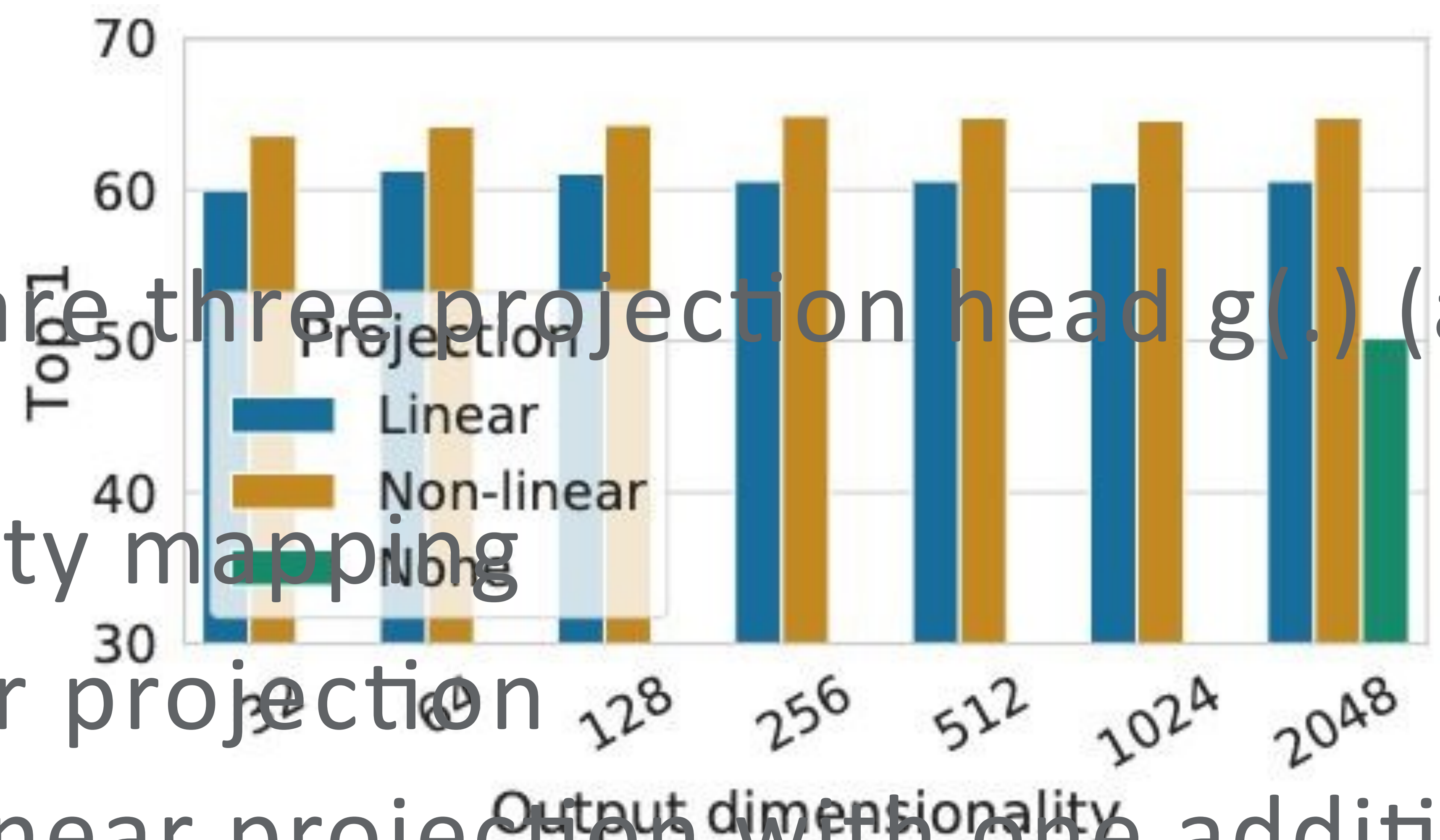


Figure 8. Linear evaluation of pretraining with different projection heads. The dimension of h (before projection) is 2048.

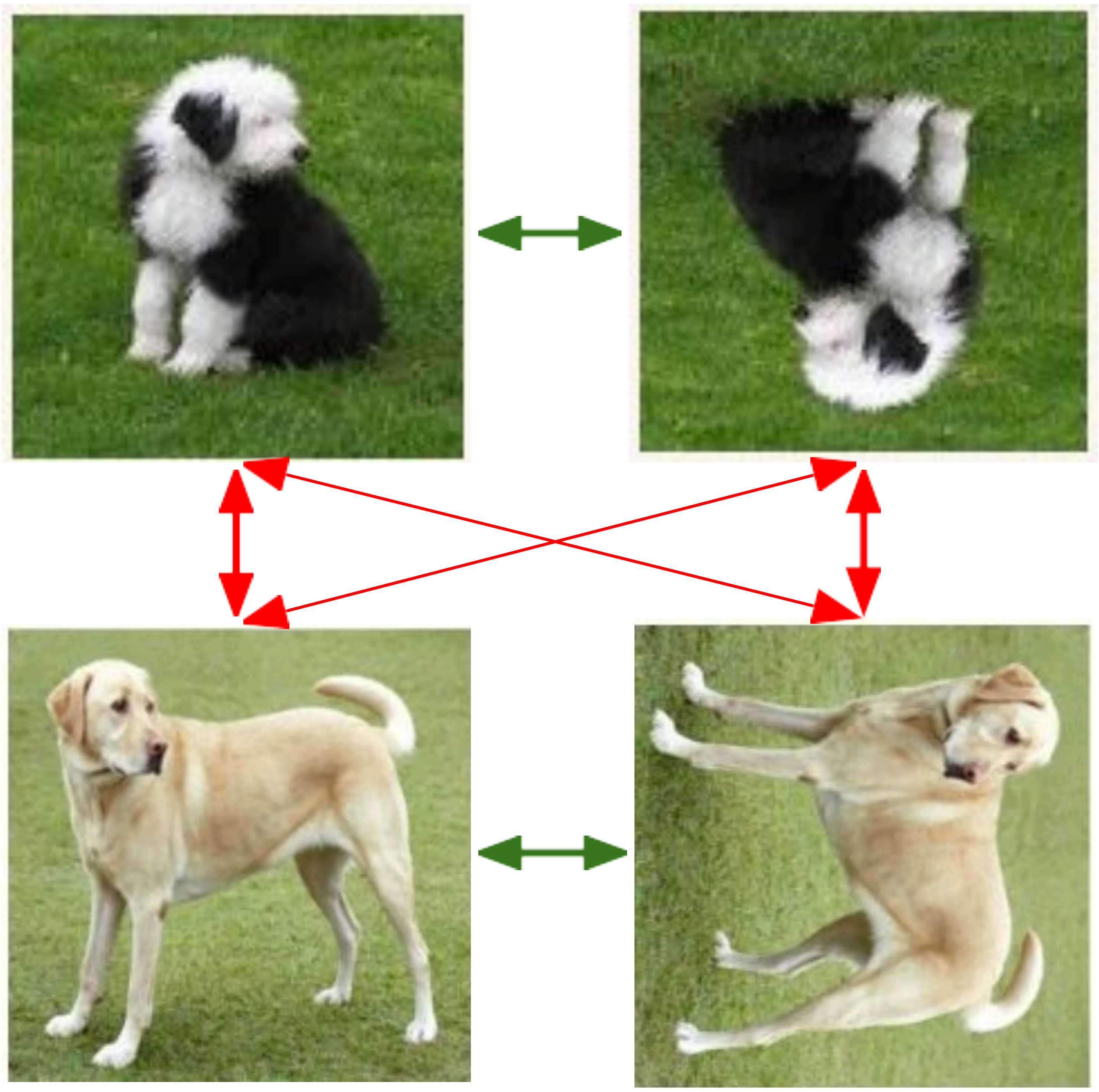
Even when non-linear projecti

A nonlinear projection head improves the representation quality of the layer before it

To understand why this happens, we measure information in ***h*** and ***z=g(h)***

What to predict?	Random guess	Representation $\boldsymbol{h}$	$g(\boldsymbol{h})$
Color vs grayscale	80	99.3	97.4
Rotation	25	67.6	25.6
Orig. vs corrupted	50	99.5	59.6
Orig. vs Sobel filtered	50	96.6	56.3

Table 3. Accuracy of training additional MLPs on different representations to predict the transformation applied. Other than crop and color augmentation, we additionally and independently add rotation (one of $\{0°, 90°, 180°, 270°\}$), Gaussian noise, and Sobel filtering transformation during the pretraining for the last three rows. Both $\boldsymbol{h}$ and $g(\boldsymbol{h})$ are of the same dimensionality, i.e. 2048.



Contrastive loss can remove/damping rotation information in the last layer when the model is asked to identify rotated variant of an image.

Loss Function and Batch Size

Normalized cross entropy loss with adjustable temperature worl

Name	Negative loss function	Gradient w.r.t. $\boldsymbol{u}$
NT-Xent	$\boldsymbol{u}^T\boldsymbol{v}^+/\tau - \log\sum_{\boldsymbol{v}\in\{\boldsymbol{v}^+,\boldsymbol{v}^-\}}\exp(\boldsymbol{u}^T\boldsymbol{v}/\tau)$	$(1-\frac{\exp(\boldsymbol{u}^T\boldsymbol{v}^+/\tau)}{Z(\boldsymbol{u})})/\tau\boldsymbol{v}^+ - \sum_{\boldsymbol{v}^-}\frac{\exp(\boldsymbol{u}^T\boldsymbol{v}^-/\tau)}{Z(\boldsymbol{u})}/\tau\boldsymbol{v}^-$

Margin	NT-Logi.	Margin (sh)	NT-Logi.(sh)	NT-Xent
50.9	51.6	57.5	57.9	63.9

Table 4. Linear evaluation (top 1) for models trained with different loss functions. "sh" means using semi-hard negative mining.

NT-Xent loss needs N and

Name	Negative loss function	Gradient w.r.t. $\boldsymbol{u}$
NT-Xent	$\boldsymbol{u}^T\boldsymbol{v}^+/\tau - \log\sum_{\boldsymbol{v}\in\{\boldsymbol{v}^+,\boldsymbol{v}^-\}}\exp(\boldsymbol{u}^T\boldsymbol{v}/\tau)$	$(1-\frac{\exp(\boldsymbol{u}^T\boldsymbol{v}^+/\tau)}{Z(\boldsymbol{u})})/\tau\boldsymbol{v}^+ - \sum_{\boldsymbol{v}^-}\frac{\exp(\boldsymbol{u}^T\boldsymbol{v}^-/\tau)}{Z(\boldsymbol{u})}/\tau\boldsymbol{v}^-$

- L2 normalization with temperature scaling makes a better loss.
- Contrastive accuracy is not correlated with linear evaluation when l2 norm and/or temperature are changed.

ℓ_2 norm?	τ	Entropy	Contrast. task acc.	Top 1
Yes	0.05	1.0	90.5	59.7
	0.1	4.5	87.8	64.4
	0.5	8.2	68.2	60.7
	1	8.3	59.1	58.0
No	10	0.5	91.7	57.2
	100	0.5	92.1	57.0

Table 5. Linear evaluation for models trained with different choices of ℓ_2 norm and temperature τ for NT-Xent loss. The contrastive distribution is over 4096 examples.

Contrastive learning benefits from larger batch sizes and

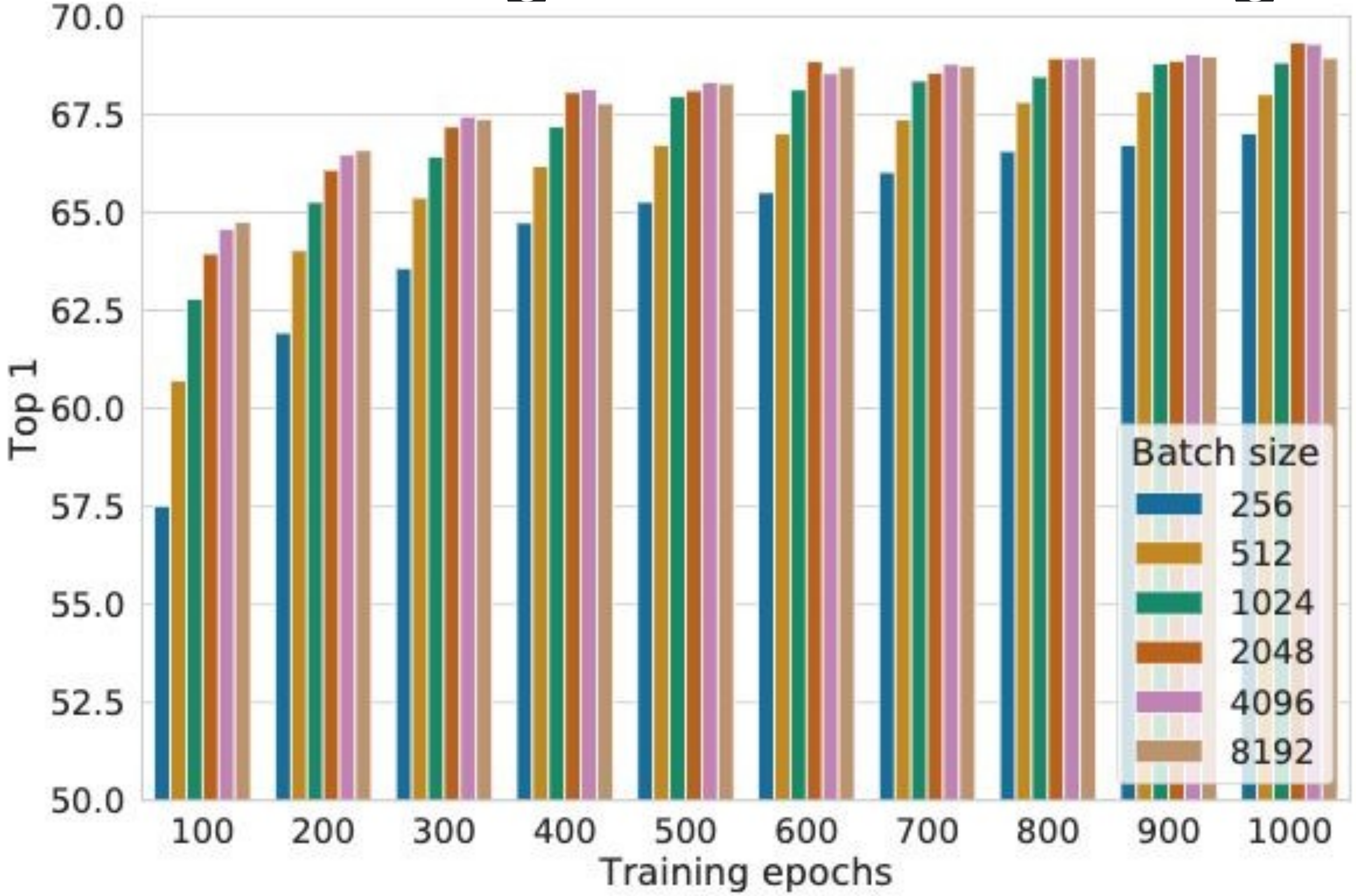


Figure 9. Linear evaluation models (ResNet-50) trained with different batch size and epochs. Each bar is a single run from scratch.

Linear evaluation

7% relative improvement over previous SOTA (cpc v2), matching fully-supervised ResNet-50.

Method	Architecture	Param.	Top 1	Top 5
Methods using ResNet-50:				
Local Agg.	ResNet-50	24	60.2	-
MoCo	ResNet-50	24	60.6	-
PIRL	ResNet-50	24	63.6	-
CPC v2	ResNet-50	24	63.8	85.3
Ours	ResNet-50	24	**69.3**	**89.0**
Methods using other architectures:				
Rotation	RevNet-50 (4×)	86	55.4	-
BigBiGAN	RevNet-50 (4×)	86	61.3	81.9
AMDIM	Custom-ResNet	626	68.1	-
CMC	ResNet-50 (2×)	188	68.4	88.2
MoCo	ResNet-50 (4×)	375	68.6	-
CPC v2	ResNet-161 (∗)	305	71.5	90.1
Ours	ResNet-50 (2×)	94	74.2	92.0
Ours	ResNet-50 (4×)	375	**76.5**	**93.2**

Table 6. ImageNet accuracies of linear classifiers trained on representations learned with different self-supervised methods.

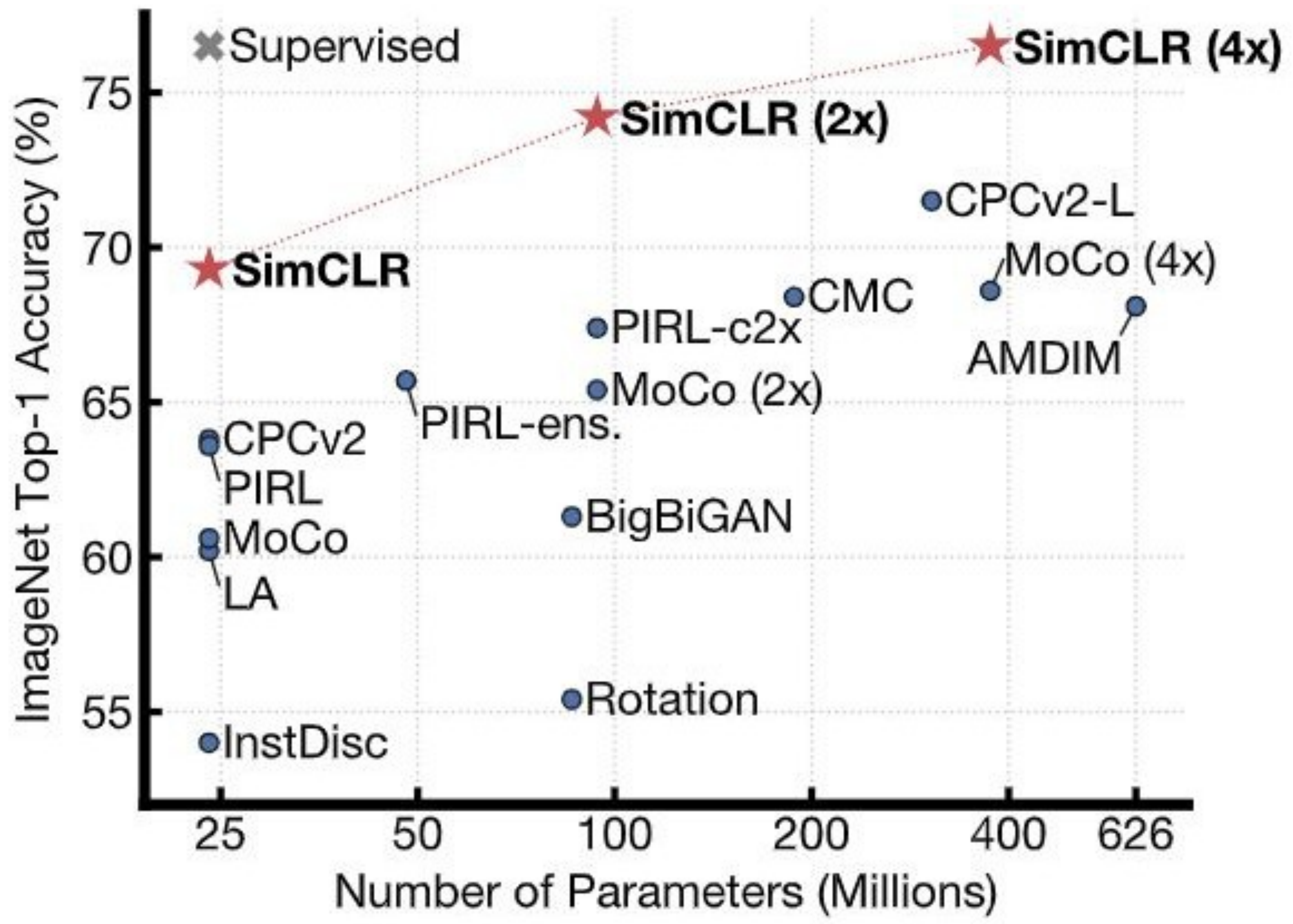


Figure 1. ImageNet top-1 accuracy of linear classifiers trained on representations learned with different self-supervised methods (pretrained on ImageNet). Our method, SimCLR, is shown in bold.

Google Research

Semi-supervised learning

10% relative improvement over previous SOTA (cpc v2), outperforms AlexNet with 100X fewer labels.

Method	Architecture	Label fraction 1%	10%
		Top 5	
Methods using other label-propagation:			
Pseudo-label	ResNet50	51.6	82.4
VAT+Entropy Min.	ResNet50	47.0	83.4
UDA (w. RandAug)	ResNet50	-	88.5
FixMatch (w. RandAug)	ResNet50	-	89.1
S4L (Rot+VAT+En. M.)	ResNet50 (4×)	-	91.2
Methods using representation learning only:			
InstDisc	ResNet50	39.2	77.4
BigBiGAN	RevNet-50 (4×)	55.2	78.8
PIRL	ResNet-50	57.2	83.8
CPC v2	ResNet-161(∗)	77.9	91.2
Ours	ResNet-50	75.5	87.8
Ours	ResNet-50 (2×)	83.0	91.2
Ours	ResNet-50 (4×)	**85.8**	**92.6**

Table 7. ImageNet accuracy of models trained with few labels.

Transfer learning

When fine-tuned, SimCLR significantly outperforms the supervised baseline on **5** datasets, whereas the supervised baseline is superior on only **2***. On the remaining **5** datasets, the models are statistically tied.

	Food	CIFAR10	CIFAR100	Birdsnap	SUN397	Cars	Aircraft	VOC2007	DTD	Pets	Caltech-101	Flowers
Linear evaluation:												
Self-supervised	**76.9**	**95.3**	80.2	48.4	**65.9**	60.0	61.2	**84.2**	**78.9**	89.2	**93.9**	**95.0**
Supervised	75.2	**95.7**	**81.2**	**56.4**	64.9	**68.8**	**63.8**	83.8	**78.7**	**92.3**	**94.1**	94.2
Fine-tuned:												
Self-supervised	**89.4**	**98.6**	**89.0**	**78.2**	**68.1**	**92.1**	**87.0**	**86.6**	**77.8**	92.1	**94.1**	97.6
Supervised	88.7	98.3	**88.7**	**77.8**	67.0	91.4	**88.0**	86.5	**78.8**	**93.2**	**94.2**	**98.0**
Random init	88.3	96.0	81.9	**77.0**	53.7	91.3	84.8	69.4	64.1	82.7	72.5	92.5

Table 8. Comparison of transfer learning performance of our self-supervised approach with supervised baselines across 12 natural image classification datasets, for ResNet-50 $(4\times)$ models pretrained on ImageNet. Results not significantly worse than the best ($p > 0.05$, permutation test) are shown in bold. See Appendix B.6 for experimental details and results with standard ResNet-50.

* The two datasets, where the supervised ImageNet pretrained model is better, are Pets and Flowers, which share a portion of labels with ImageNet.

Conclusion

- SimCLR is a simple yet effective self-supervised learning framework, advancing state-of-the-art by a large margin.
- The superior performance of SimCLR is not due to any *single* design choice, but *a combination of* design choices.
- Our studies reveal several important factors that enable effective representation learning, which could help future research.

Code & checkpoints available in github.com/google-research/simclr.

References

- Dosovitskiy, A., Fischer, P., Springenberg, J. T., Riedmiller, M., & Brox, T. Discriminative unsupervised feature learning with exemplar convolutional neural networks, arXiv preprint. *arXiv preprint arXiv:1506.02753*.

- Bojanowski, P., & Joulin, A. Unsupervised Learning by Predicting Noise. *arXiv preprint arXiv:1704.05310*.

- Pathak, D., Girshick, R., Dollár, P., Darrell, T., & Hariharan, B. Learning Features by Watching Objects Move. *In CVPR 2017*.

References

- **Colorization:**
- Larsson, G., Maire, M., & Shakhnarovich, G. Learning representations for automatic colorization. In *ECCV 2016*.
- Larsson, G., Maire, M., & Shakhnarovich, G. Colorization as a Proxy Task for Visual Understanding. *In CVPR 2017*.
- **Optical Flow**
- J. J. Yu, A. W. Harley, and K. G. Derpanis. Back to Basics: Unsupervised Learning of Optical Flow via Brightness Constancy and Motion Smoothness. In ECCVW, 2016.
- Zhu, Y., Lan, Z., Newsam, S., & Hauptmann, A. G. Guided optical flow learning. *arXiv preprint arXiv:1702.02295*.
- Ren, Z., Yan, J., Ni, B., Liu, B., Yang, X., & Zha, H. Unsupervised Deep Learning for Optical Flow Estimation. In *AAAI* 2017
- **Others**
- Cruz, R. S., Fernando, B., Cherian, A., & Gould, S. DeepPermNet: Visual Permutation Learning. *arXiv preprint arXiv:1704.02729*.
- Nair, A., Chen, D., Agrawal, P., Isola, P., Abbeel, P., Malik, J., & Levine, S. Combining Self-Supervised Learning and Imitation for Vision-Based Rope Manipulation. *arXiv preprint arXiv:1703.02018*.
- Pinto, L., Gandhi, D., Han, Y., Park, Y. L., & Gupta, A. The curious robot: Learning visual representations via physical interactions. In *ECCVW 2016*.

Bonus

What happens when we "super-size" self-supervised learning?

Scaling and benchmarking self-supervised visual representation learning -

Goyal, Mahajan, Gupta*, Misra* - 2019 https://arxiv.org/abs/1905.01235

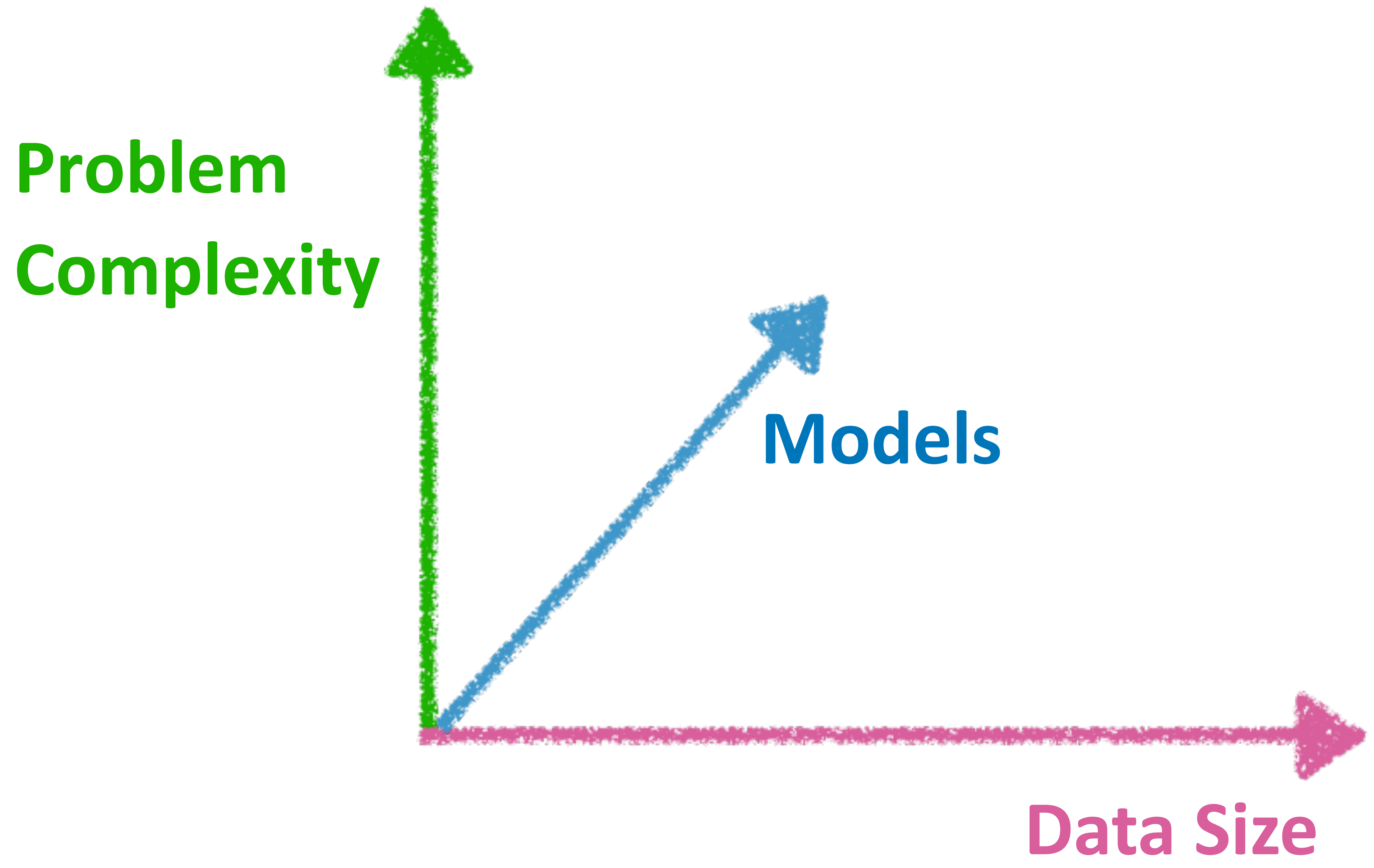
Bits of information

- For ImageNet we have ~1M images and 1000 classes
  - Each image has $\log_2(1000)$ bits of information
  - Total = 1M x $\log_2(1000)$

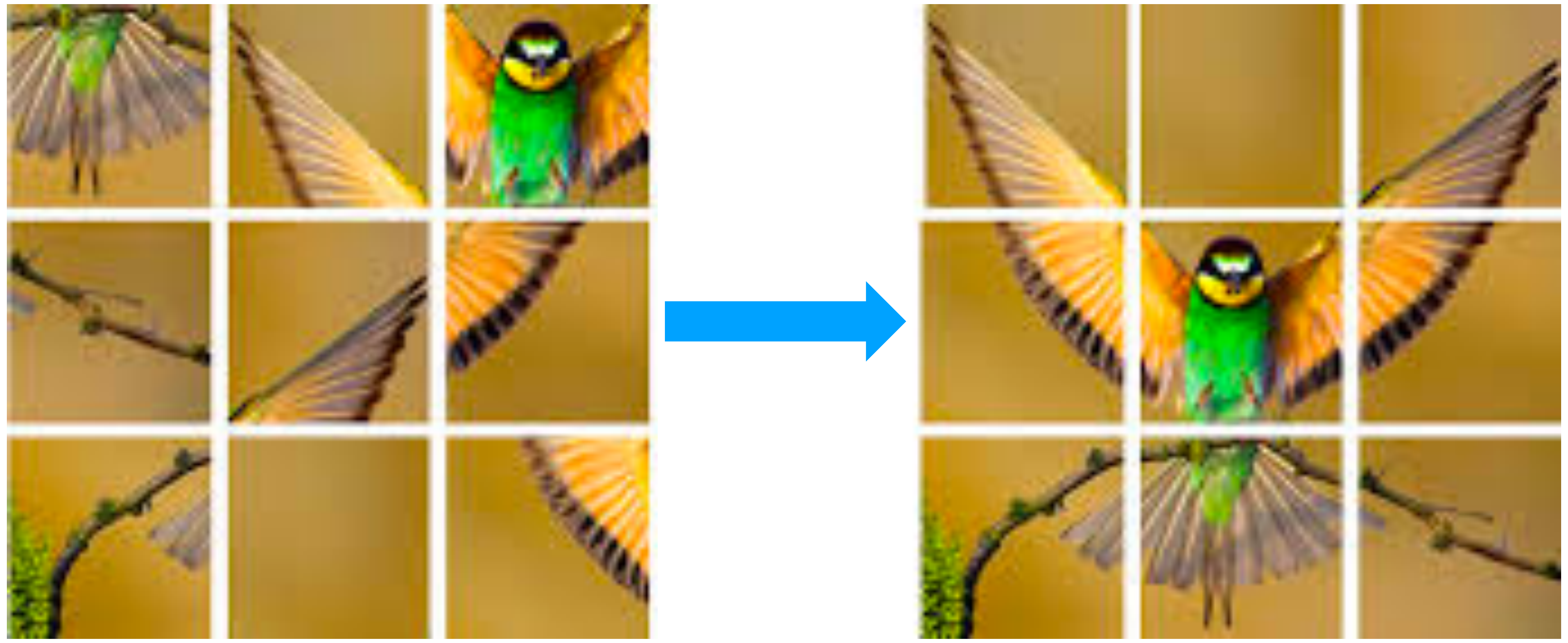
- For self-supervised methods we have ~1M images
  - Each image has lesser bits of information - $\log_2(B)$
  - Total = 1M x $\log_2(B)$
  - **Increase 1M to 100M?**
  - **Increase B to 10B?**

How large scale?

- Focus on existing **popular image-based** self-supervised methods
- Scale along three axes

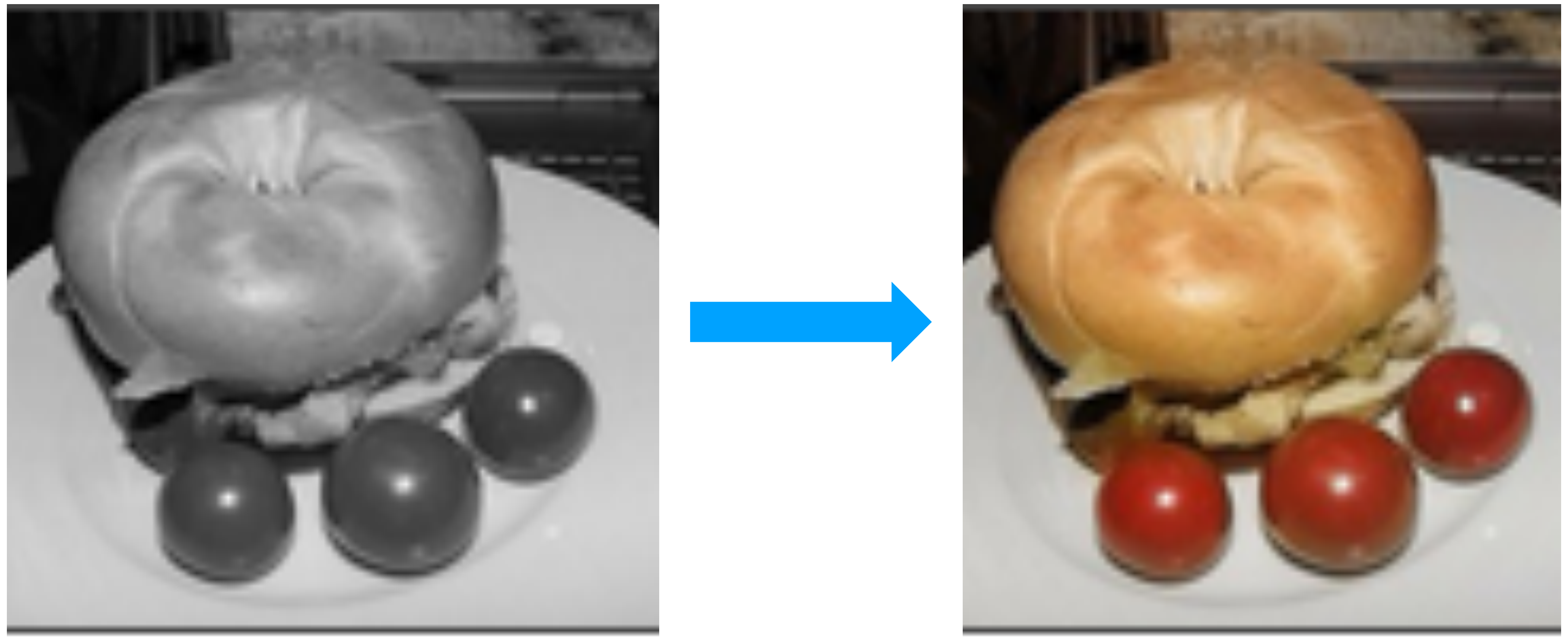


Specific self-supervision problems



Jigsaw puzzles

(Noorozi & Favaro, 2016)

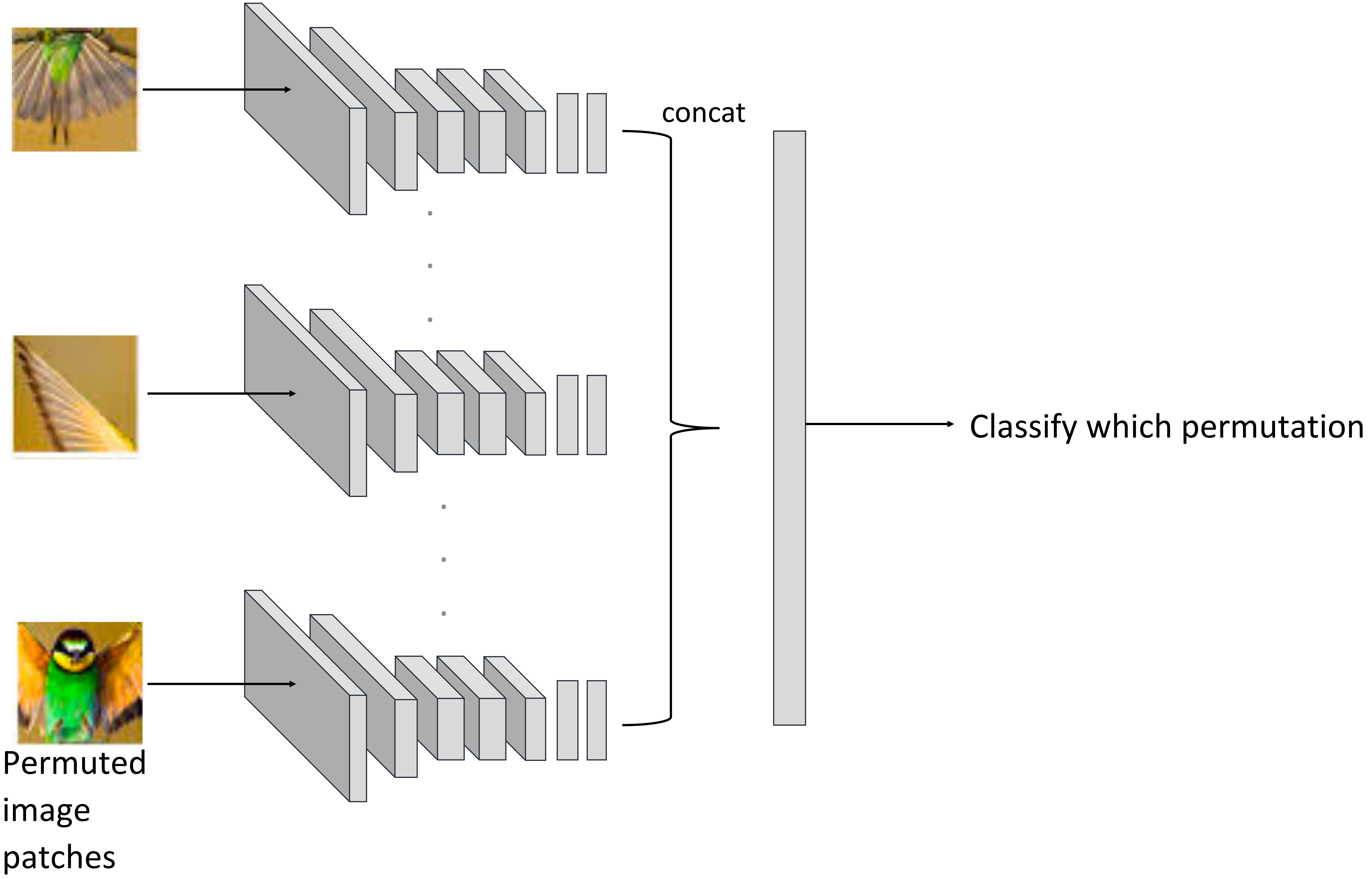


Colorization

(Zhang & Efros, 2016)

Images from the ImageNet dataset

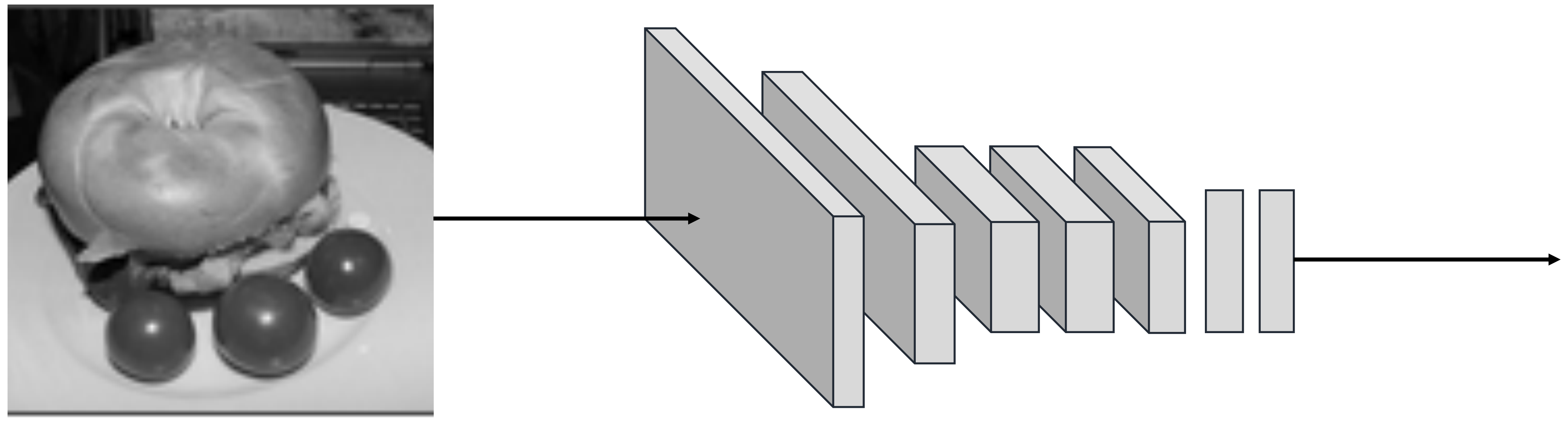
Jigsaw Puzzles



- Use N=9 patches
- In practice, use a subset of permutations
- E.g. 100 from 9!
- Each patch is processed independently
- N-way ConvNet (shared params)

- Problem Complexity
  - Size of subset

Colorization

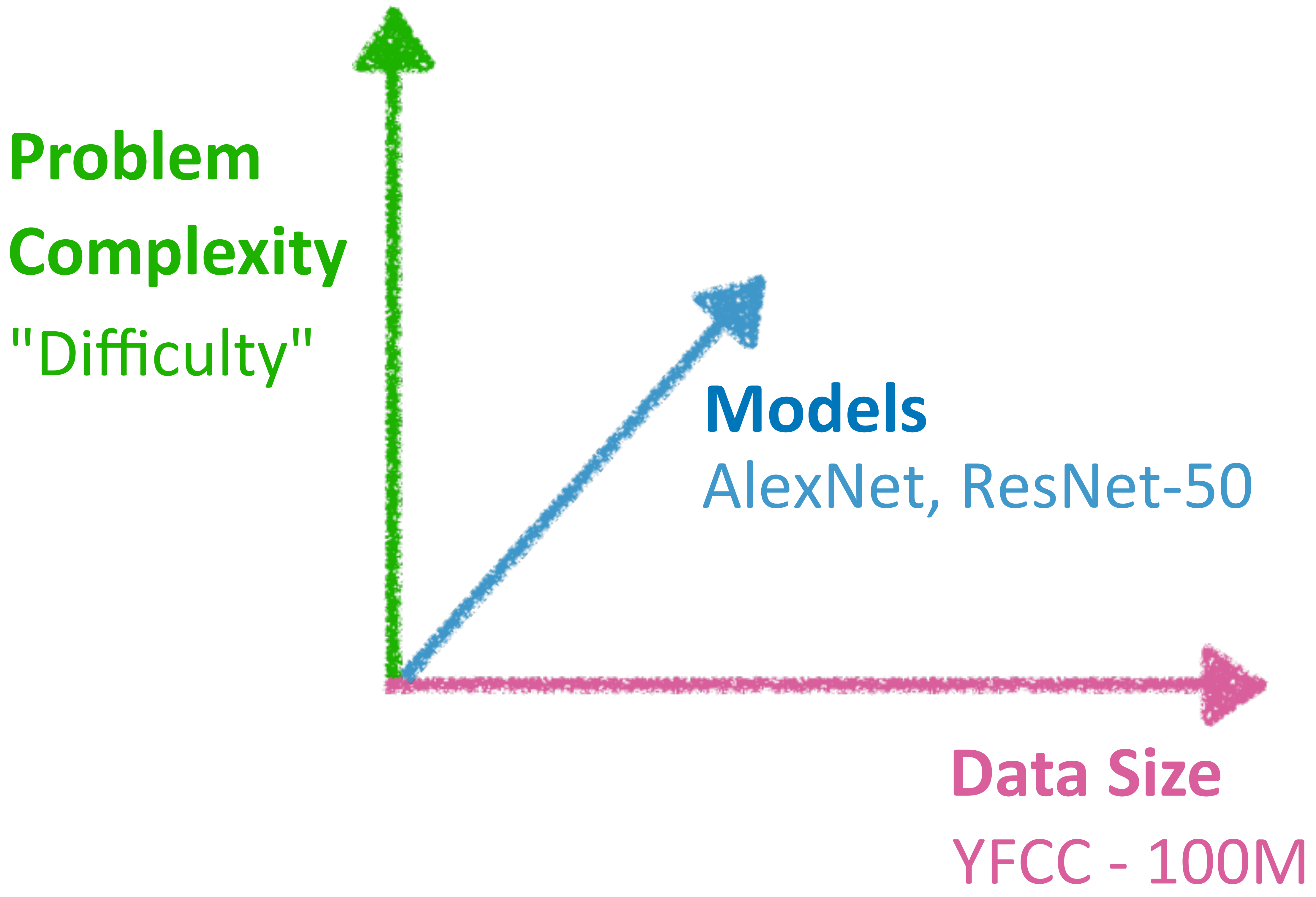


At each pixel, classify which color bins

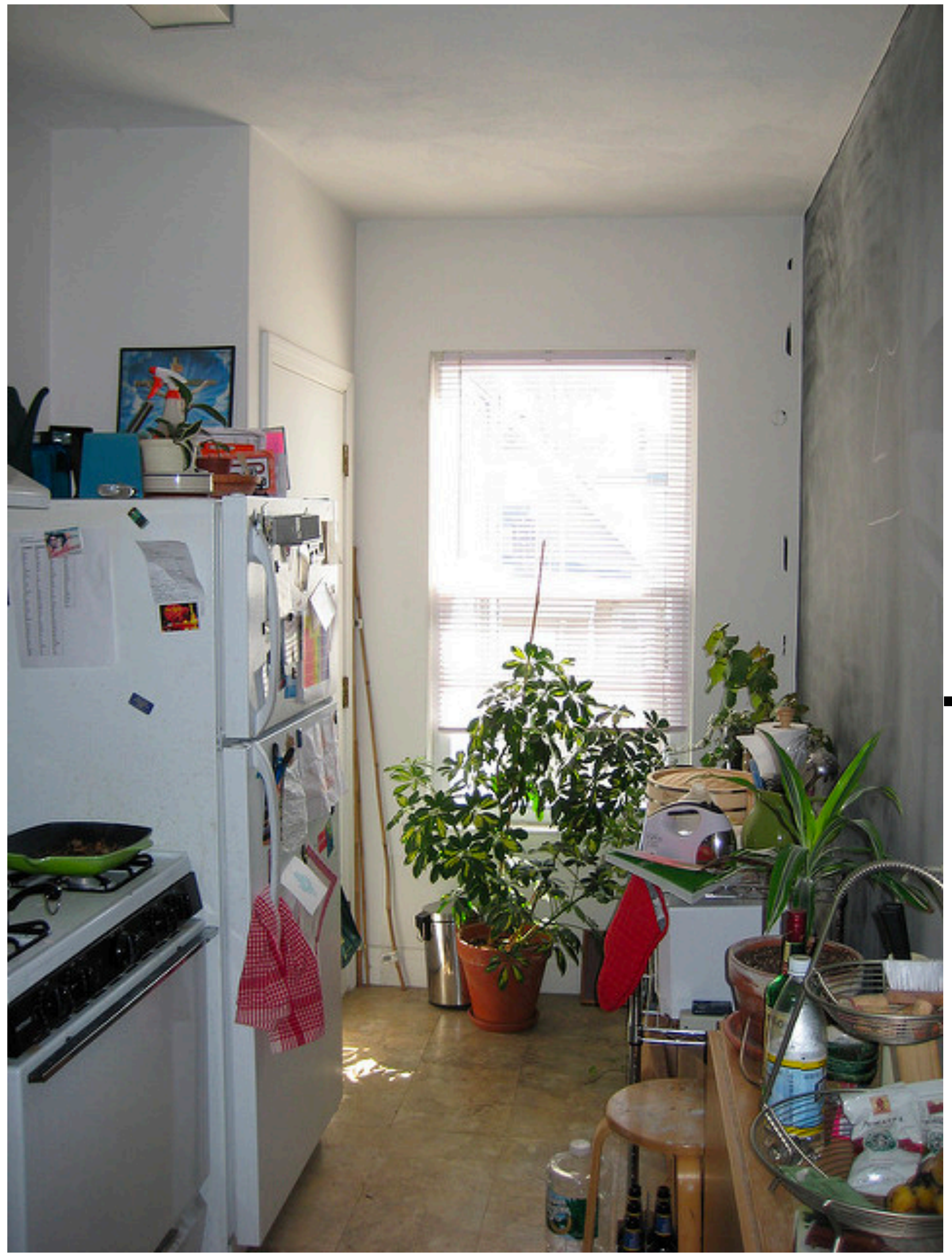
- Predict N=313 colors
  - Binning LAB space
  - Multi-modal loss
- Problem Complexity
  - Number of colors
  - Number of neighbors in multi-modal loss

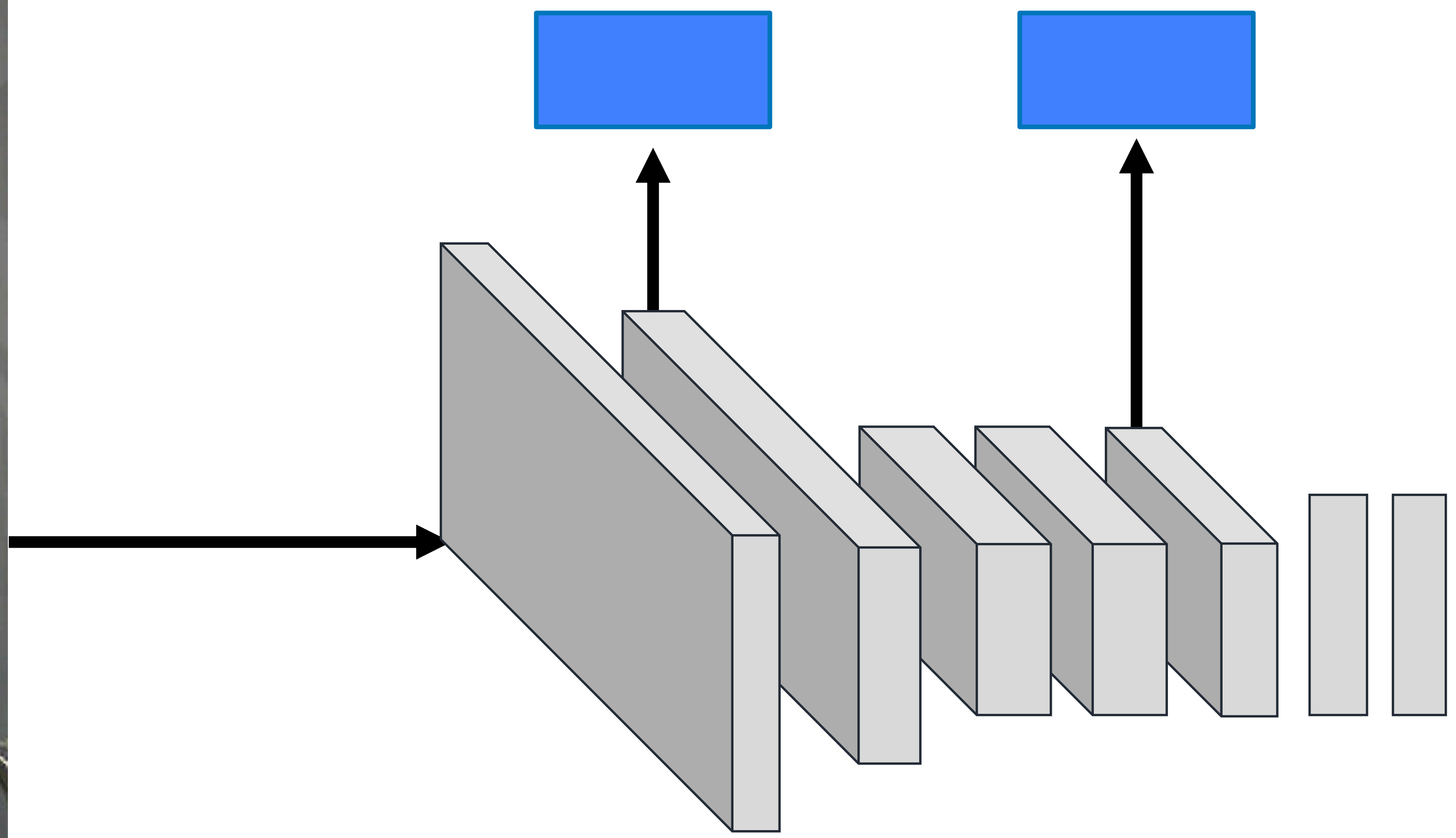
How large scale?

- Scale two techniques - Jigsaw and Colorization
- Scale along three axes



Evaluating the representation





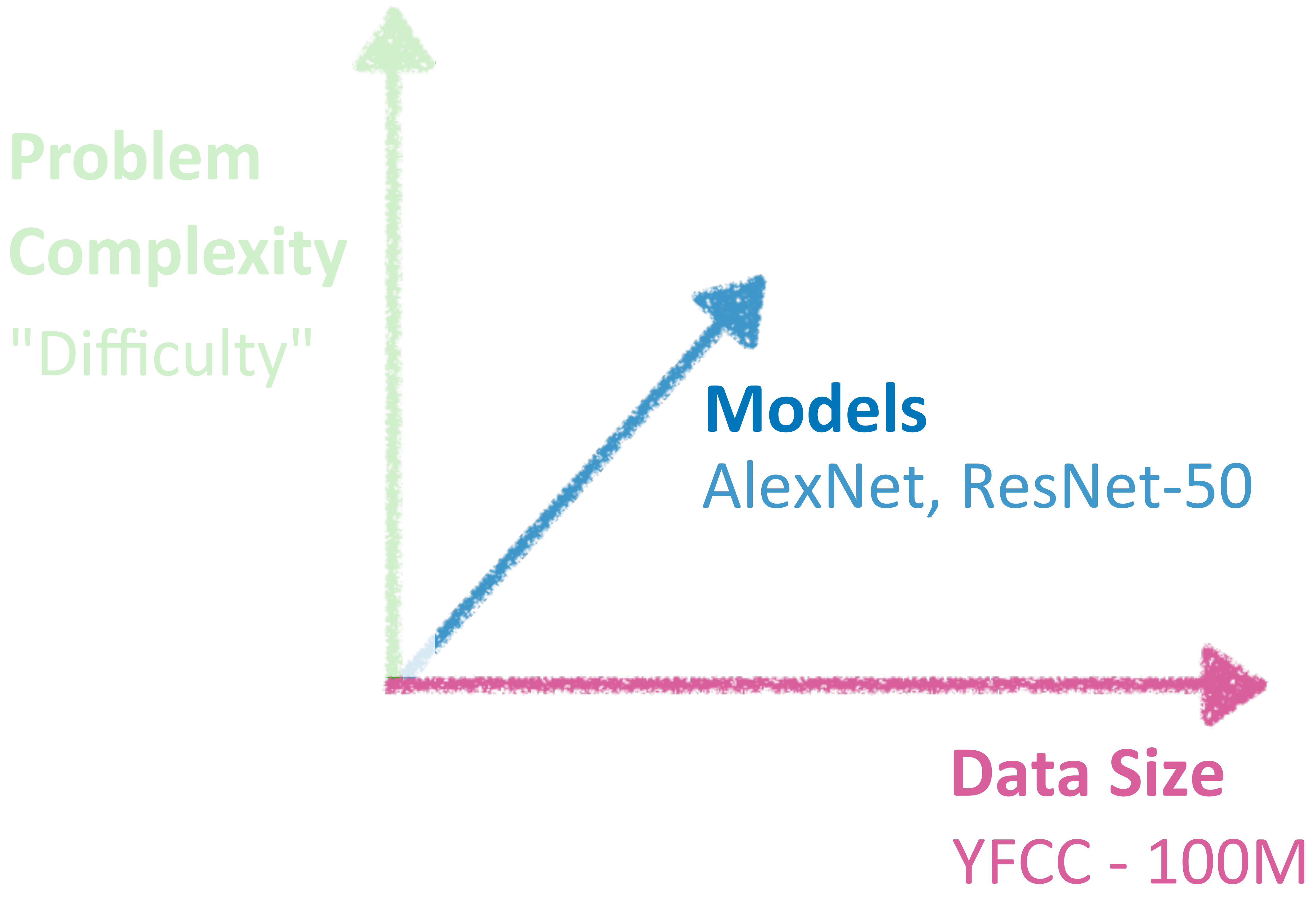
"Investigation" task

- Train a Linear SVM on **fixed feature** representations

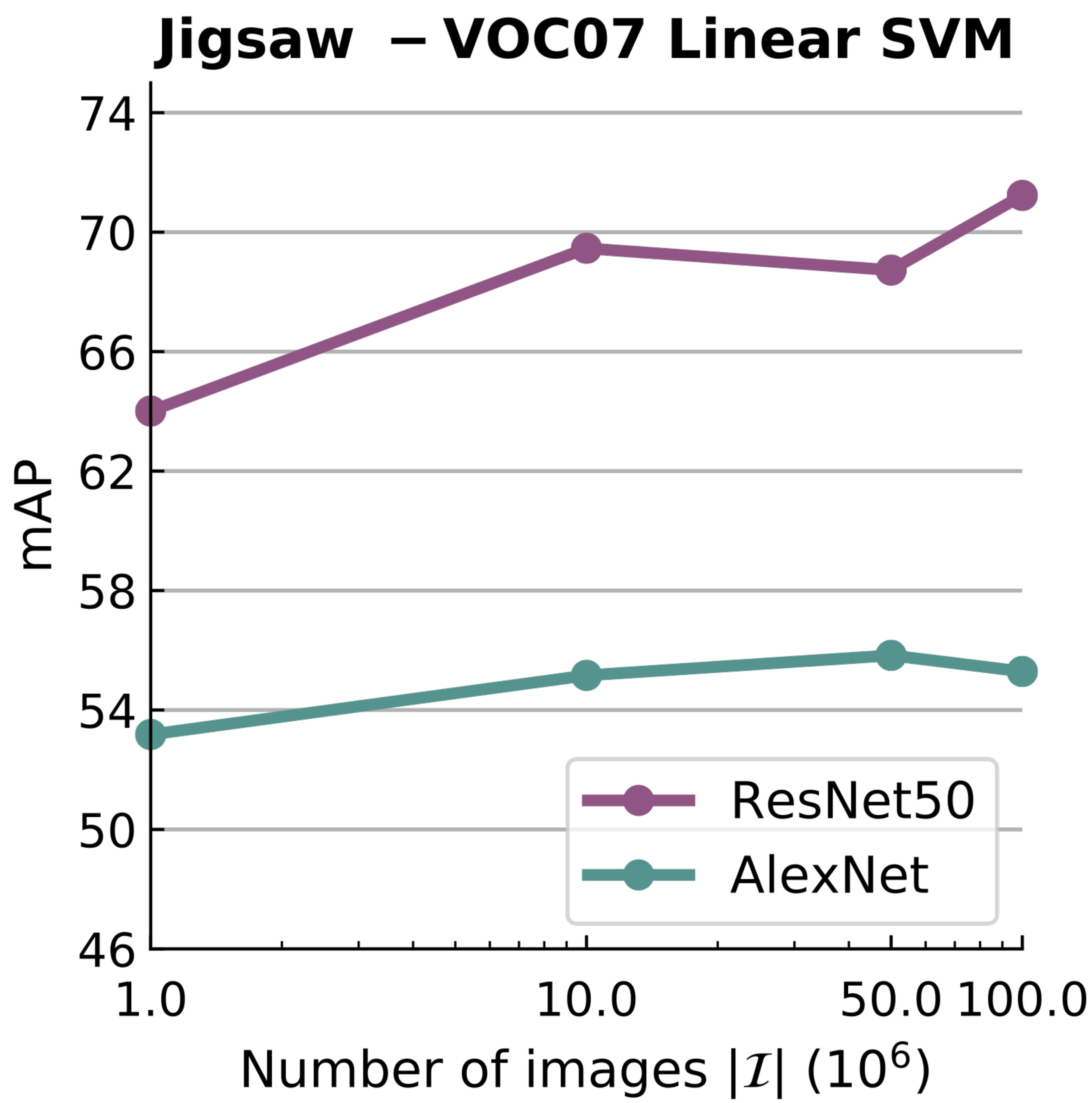
"Investigation" task

- Train a Linear SVM on **fixed feature** representations
- Use the VOC07 image classification task

Scaling on Data Axis



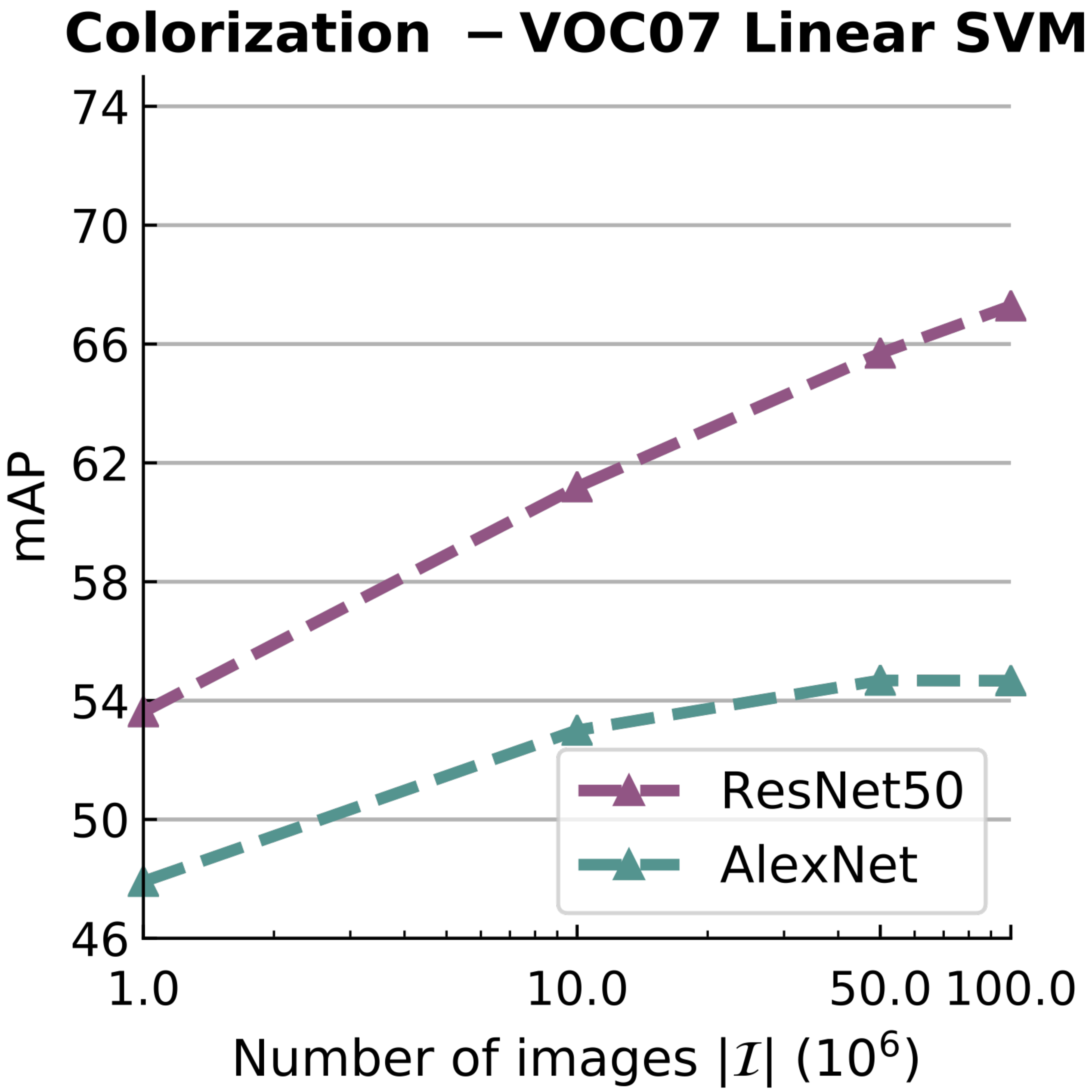
Scaling on Data Axis



Gain for ResNet50: **10** points

Gain for AlexNet: 2 points

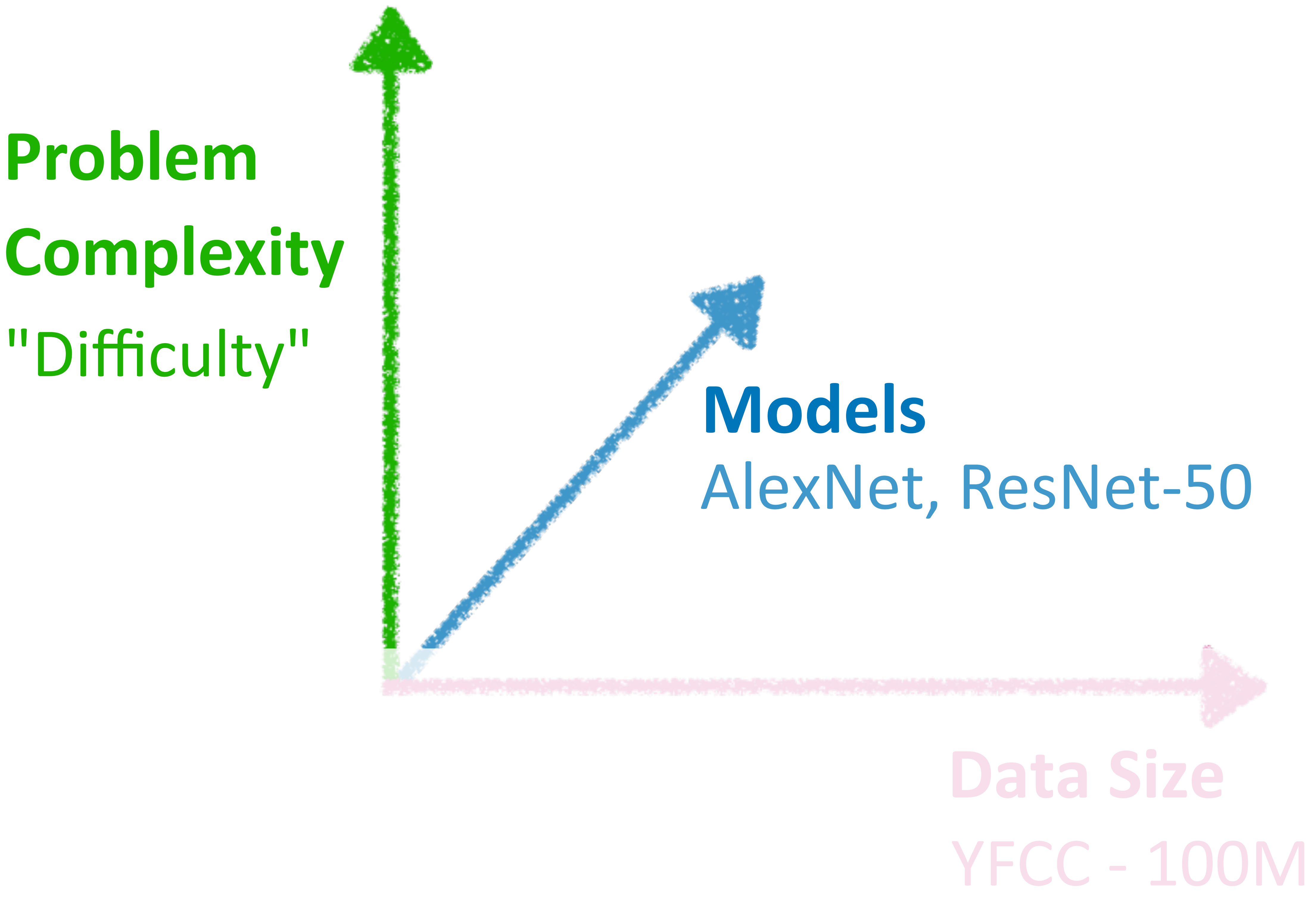
Scaling on Data Axis



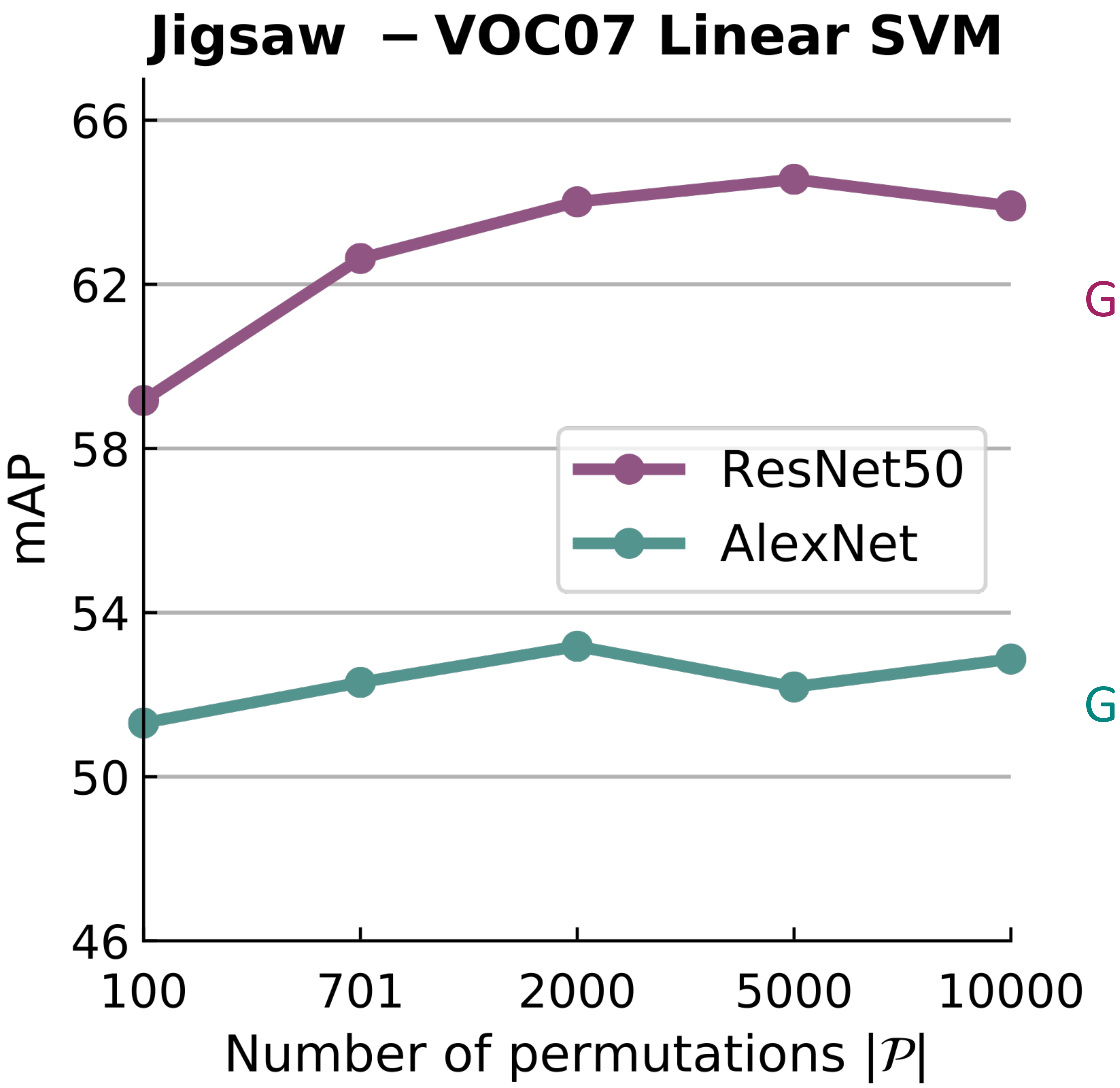
Gain for ResNet50: **12** points

Gain for AlexNet: 8 points

Scaling on Problem Complexity



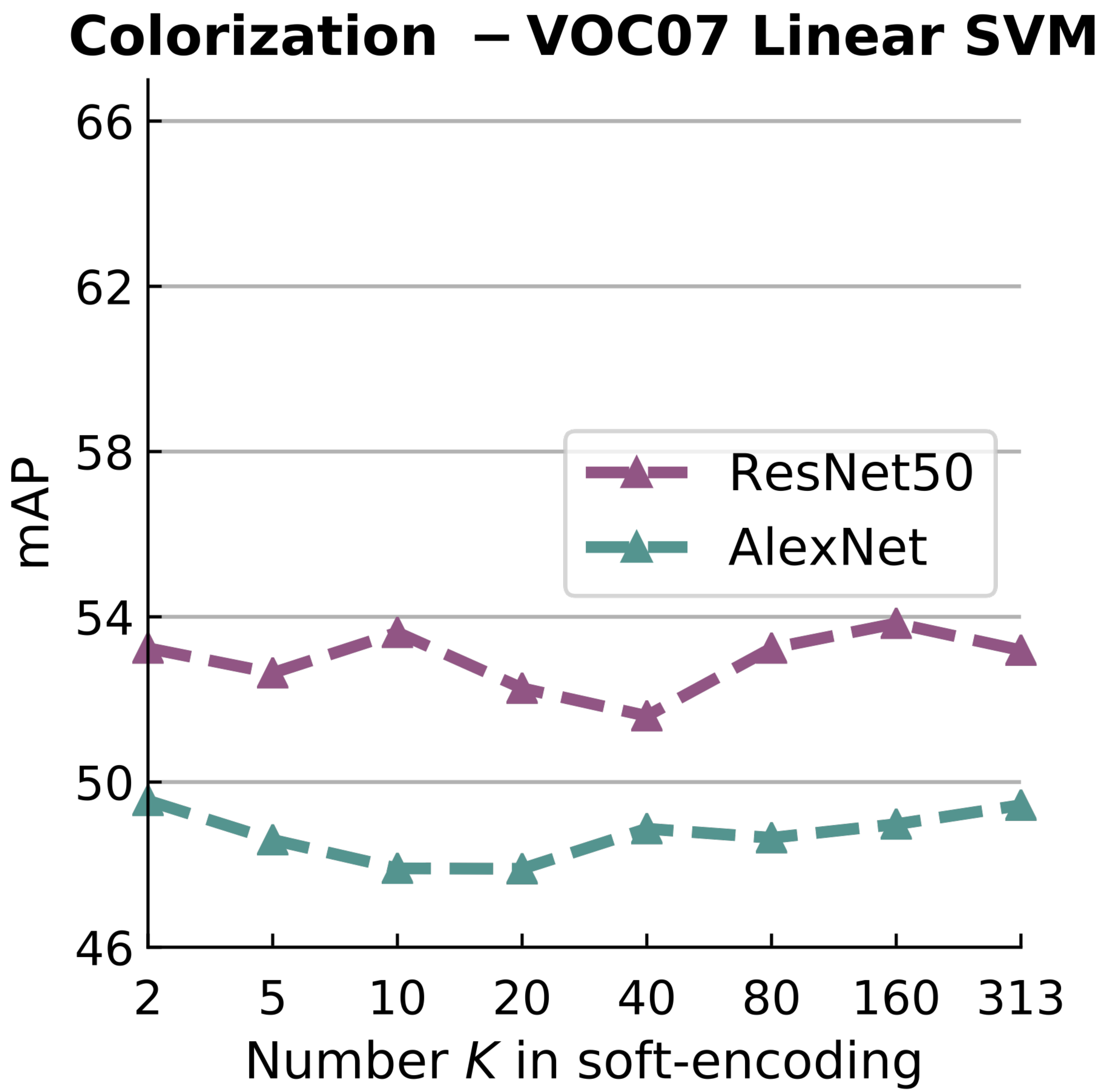
Scaling on Problem Complexity Axis



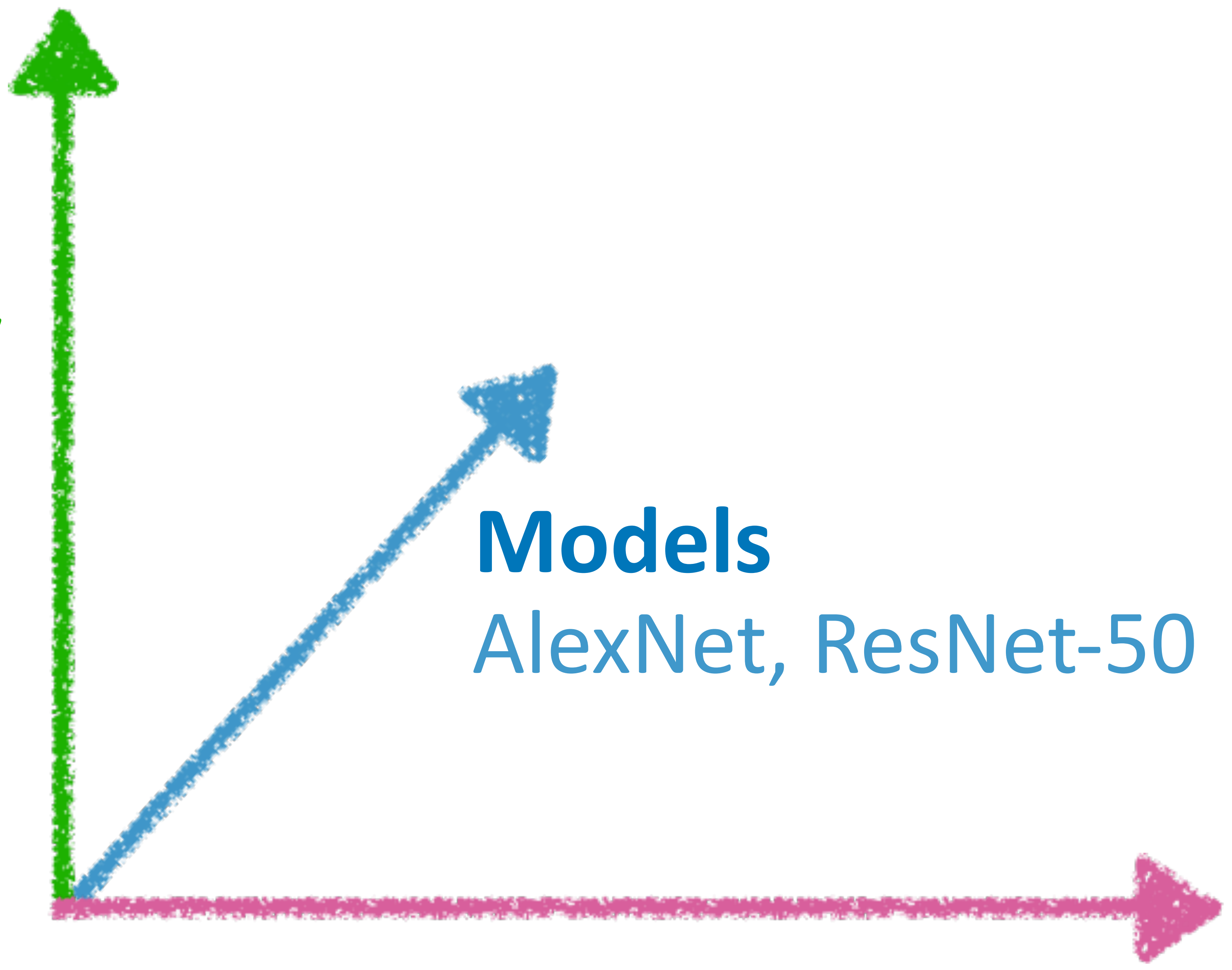
Gain for ResNet50: **6** points

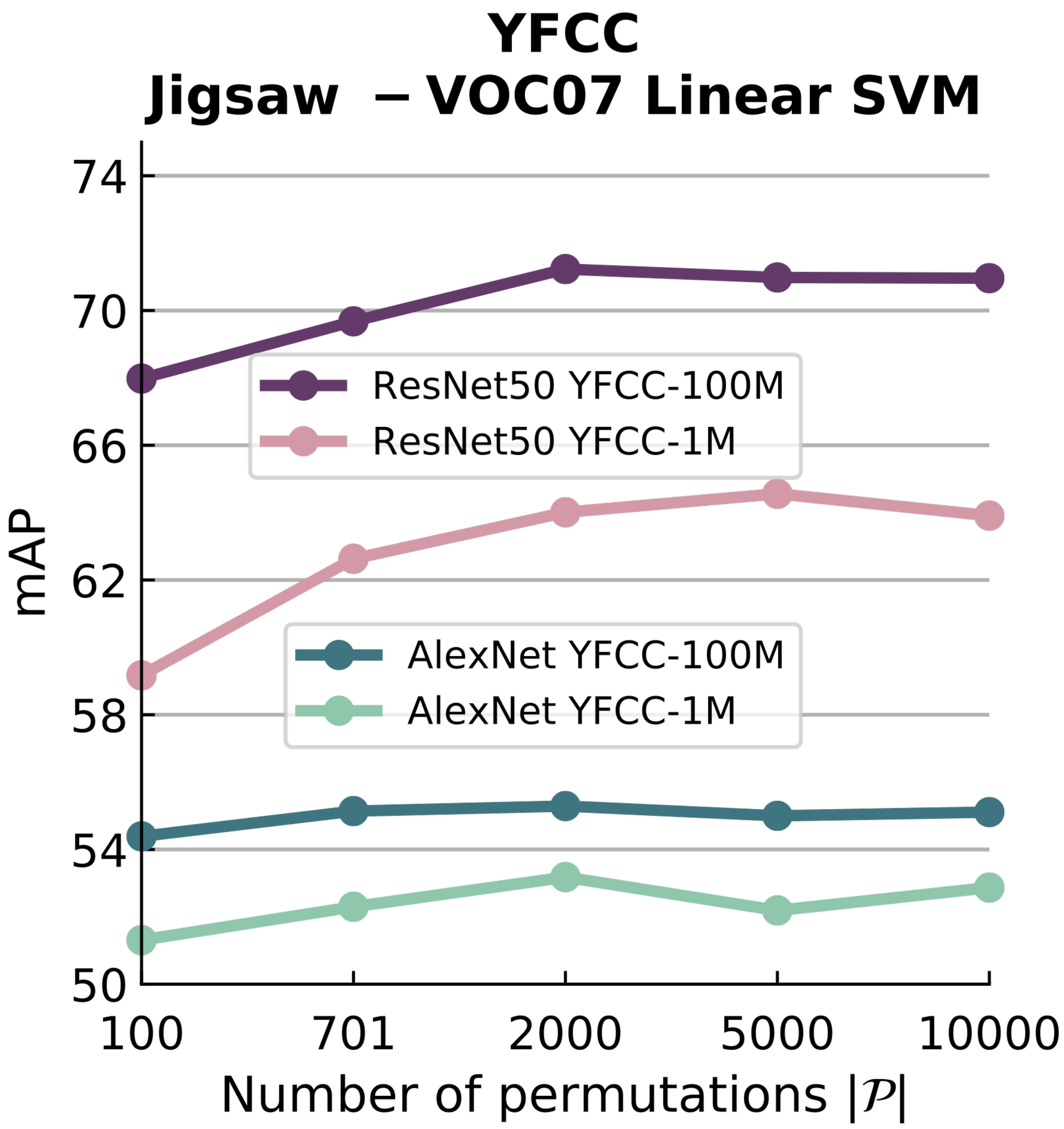
Gain for AlexNet: 2 points

Scaling on Problem Complexity Axis



Scaling on Data and Problem





Gains along **both** data and problem
axes are **complementary**

Our Evaluation – many tasks

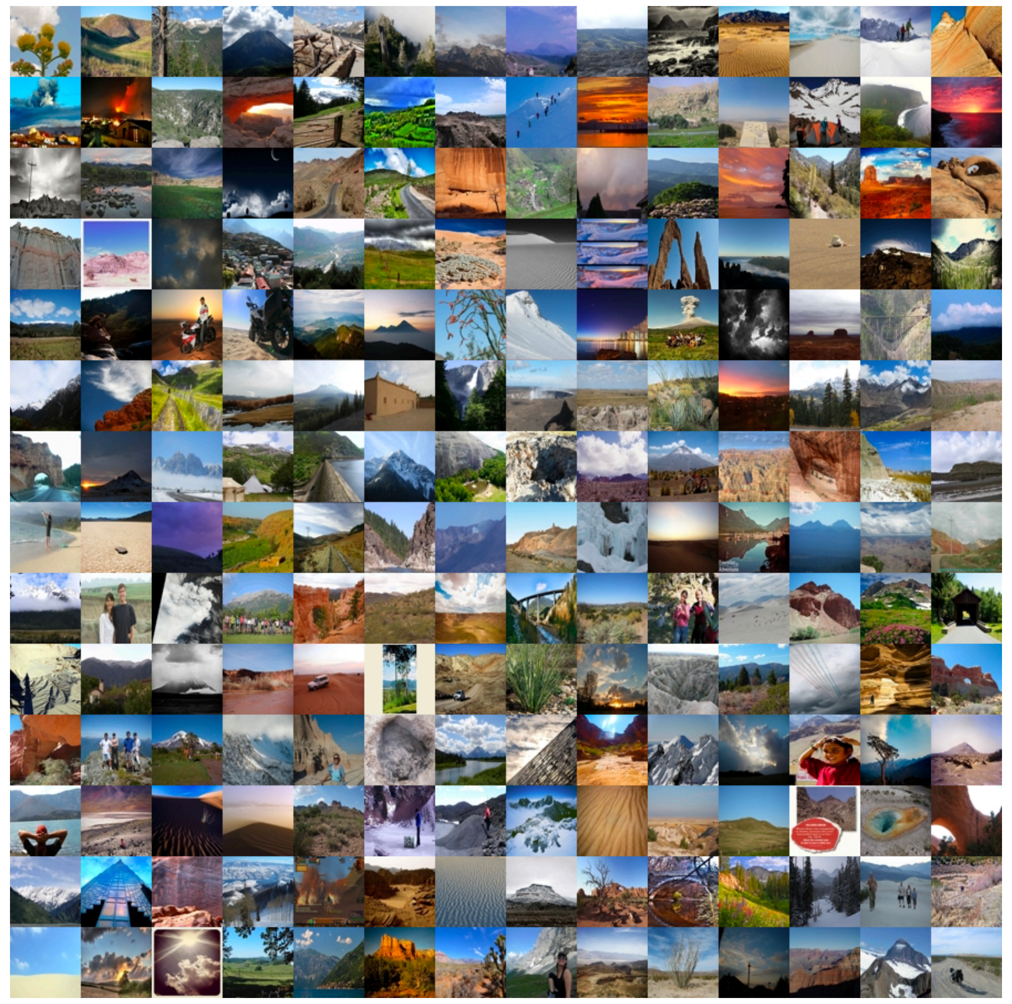
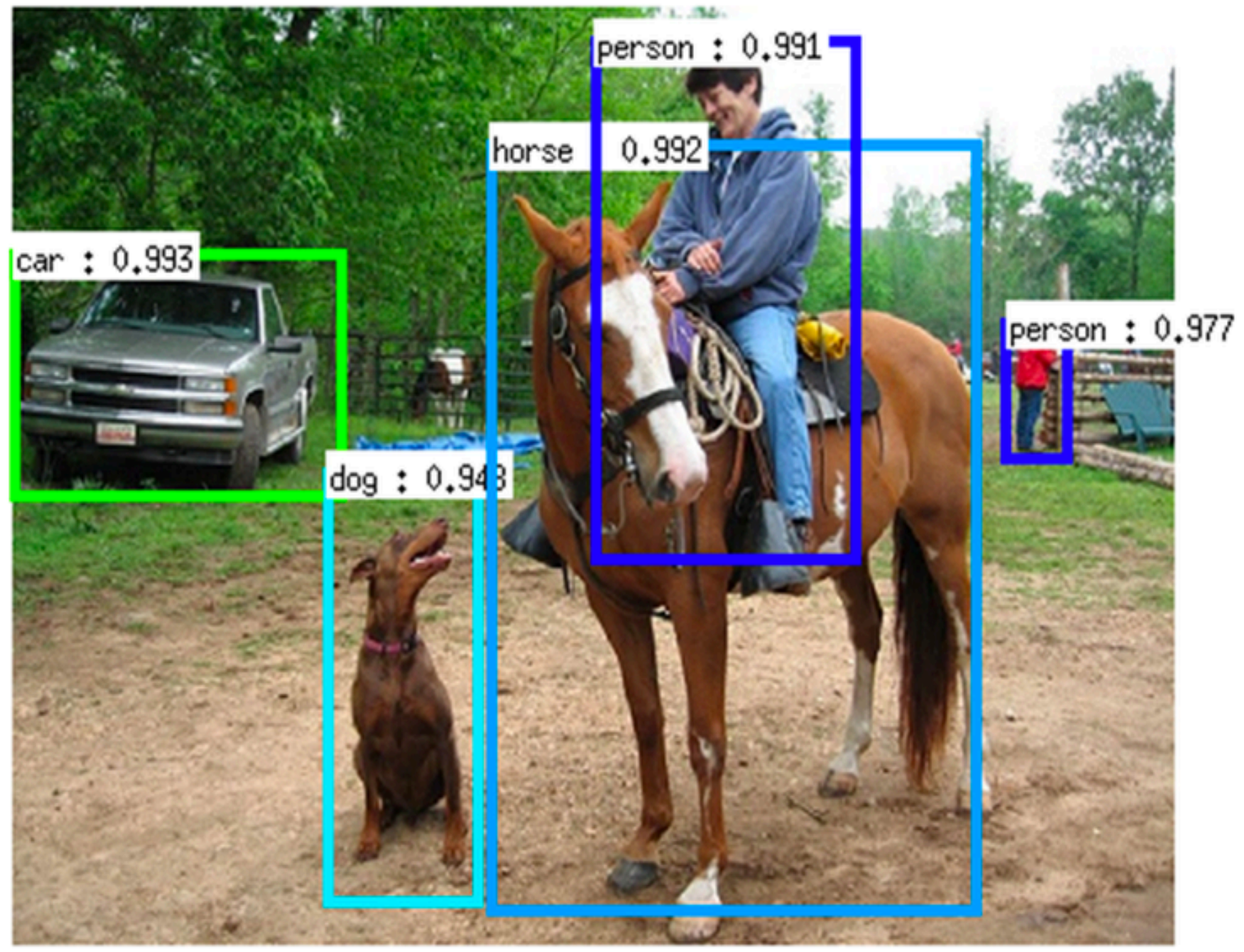


Image classification

Few-shot learning

ImageNet, Places-205, VOC’07, COCO



Object detection

VOC’07

3D Understanding

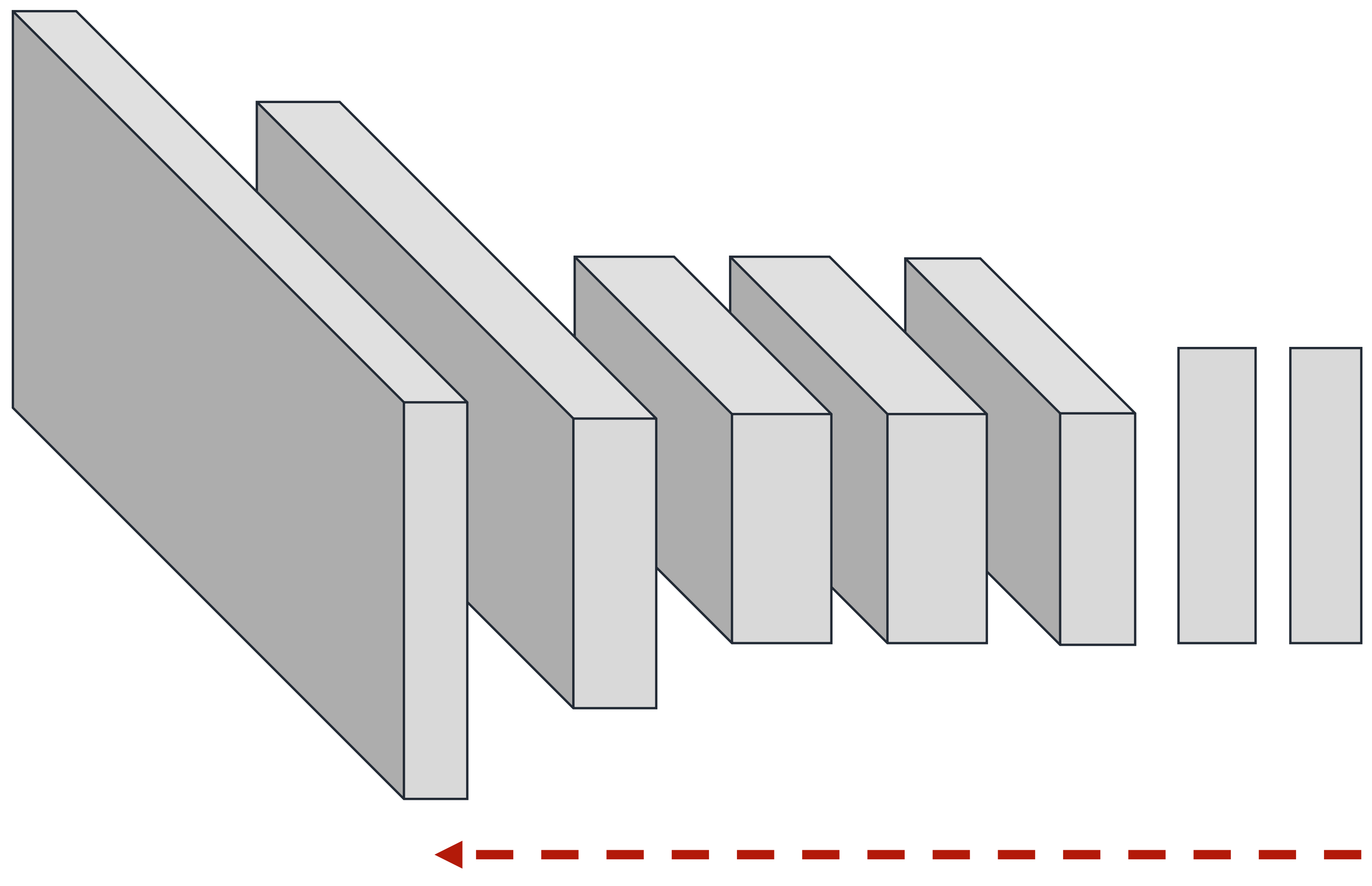
Surface Normals – NYUv2

Navigation

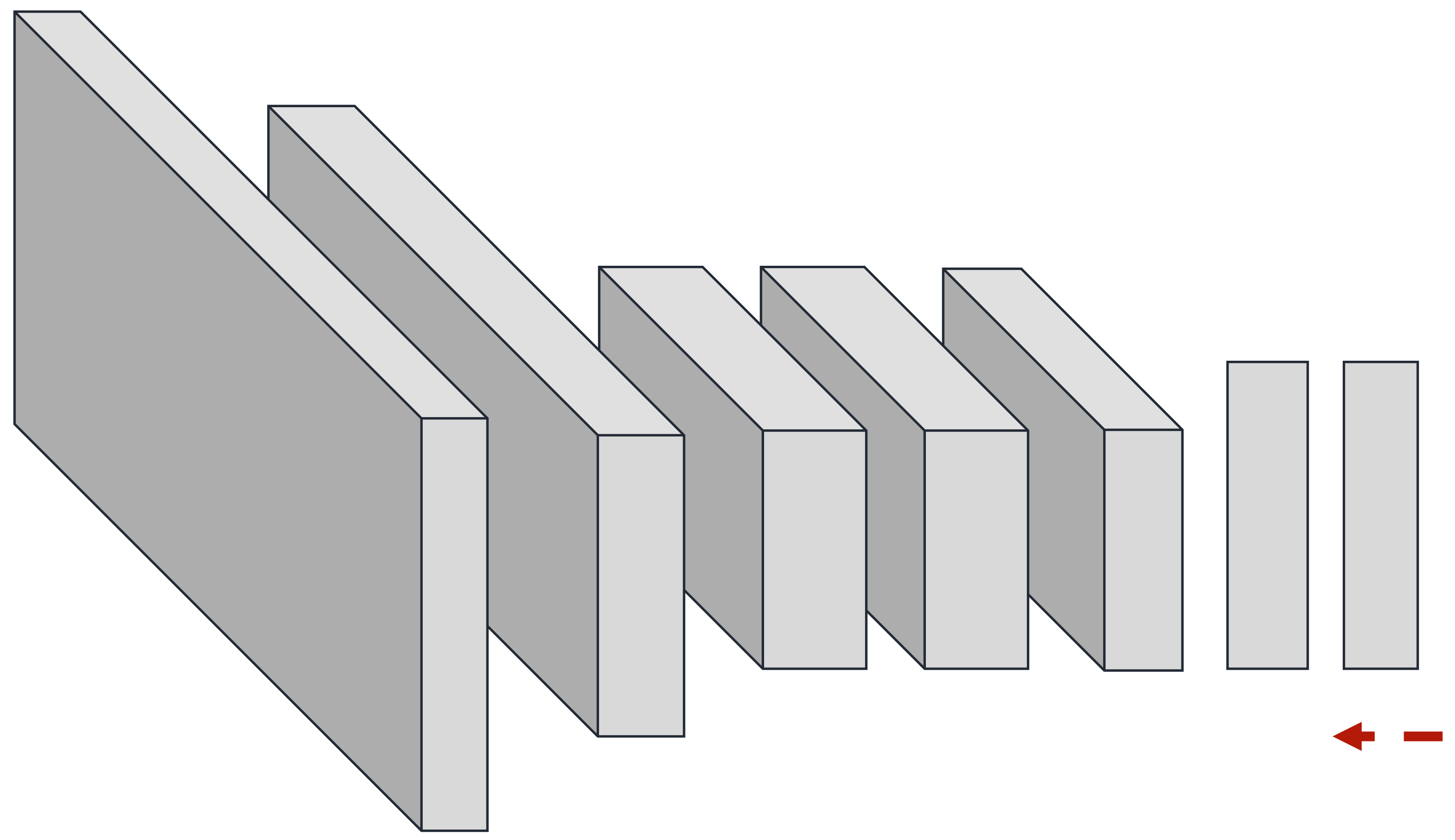
Gibson environment

Images from the Places, VOC07, NYUv2 and Gibson datasets

Our Evaluation – fine-tuning vs. linear classifier



Fine-tune all layers



Linear classifier

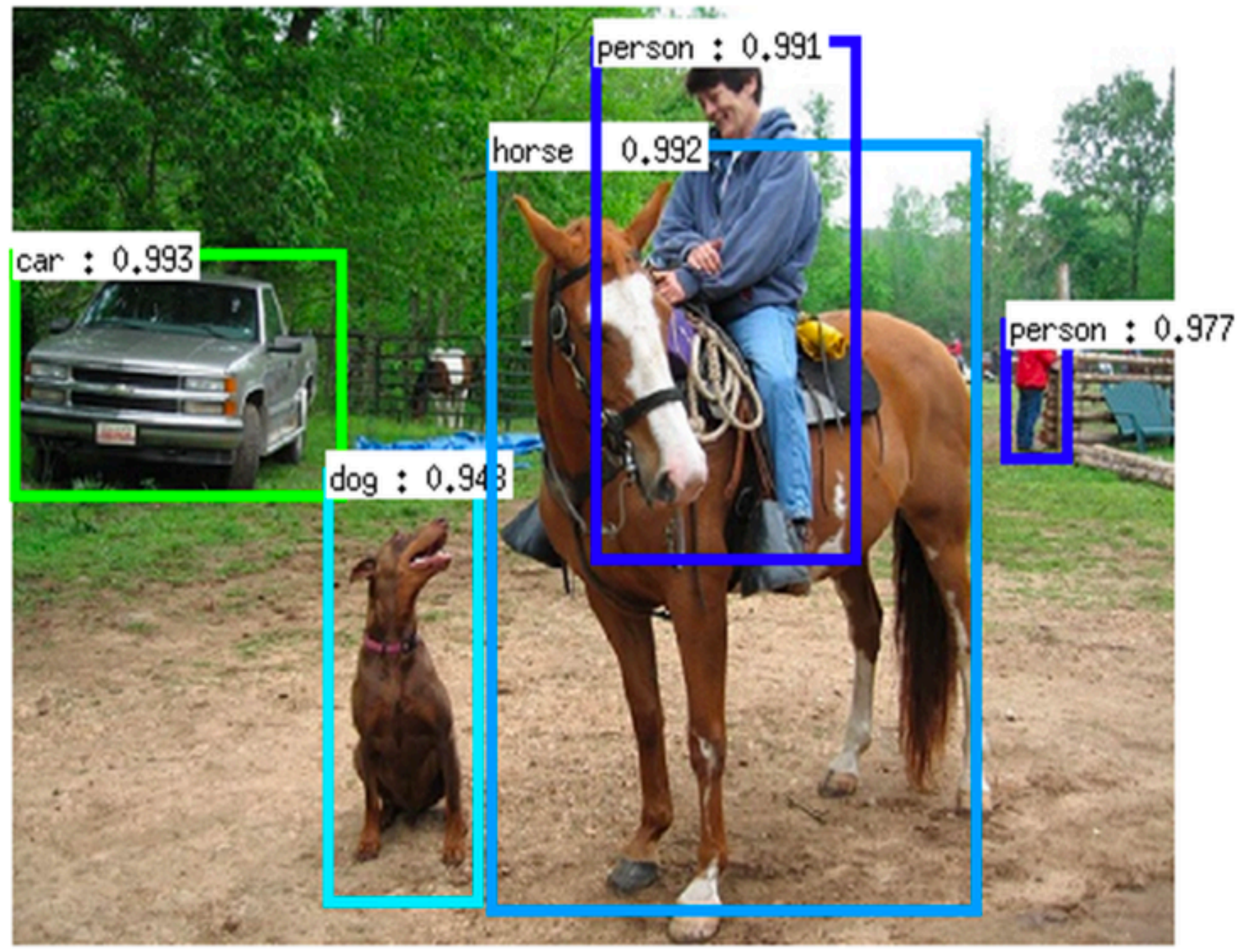
*A good representation transfers with **little training***

Object Detection

Image classification

Few-shot learning

ImageNet, Places-205, VOC'07, COCO



Object detection

VOC'07

3D Understanding

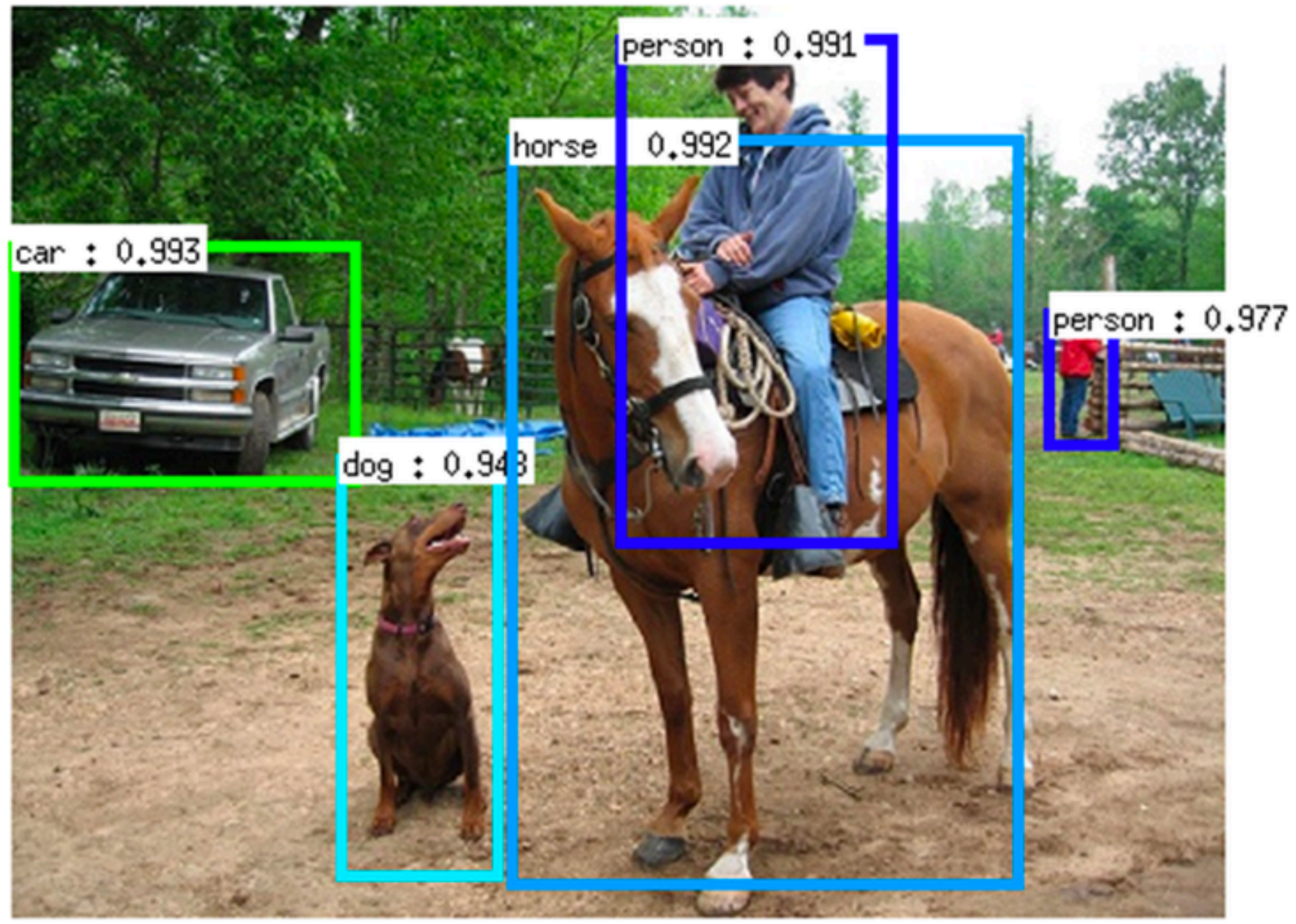
Surface Normals – NYUv2

Navigation

Gibson environment

Object Detection

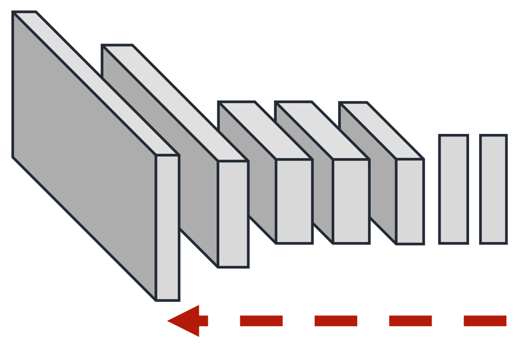
- Fast R-CNN (Girshick et al., 2015)
  - Same optimization parameters for all methods (including supervised)
  - No “bells and whistles”
  - Use VOC’07



Object detection

Image from the VOC07 dataset

Object Detection



Fine-tune all

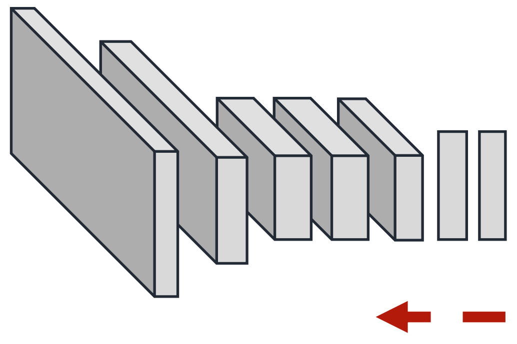
VOC07 test set.

Fast R-CNN ResNet50

Initialization	Train Set	
	VOC07	VOC07+12
ImageNet Supervised	70.5	76.2
Places Supervised	67.2	74.5
Jigsaw ImageNet-1k	61.4	68.3
Jigsaw ImageNet-22k	69.2	75.4
Jigsaw YFCC100M	66.6	73.3

within error

Object Detection - Training RoI heads only



Last layer

VOC07 test set.

Fast R-CNN ResNet50

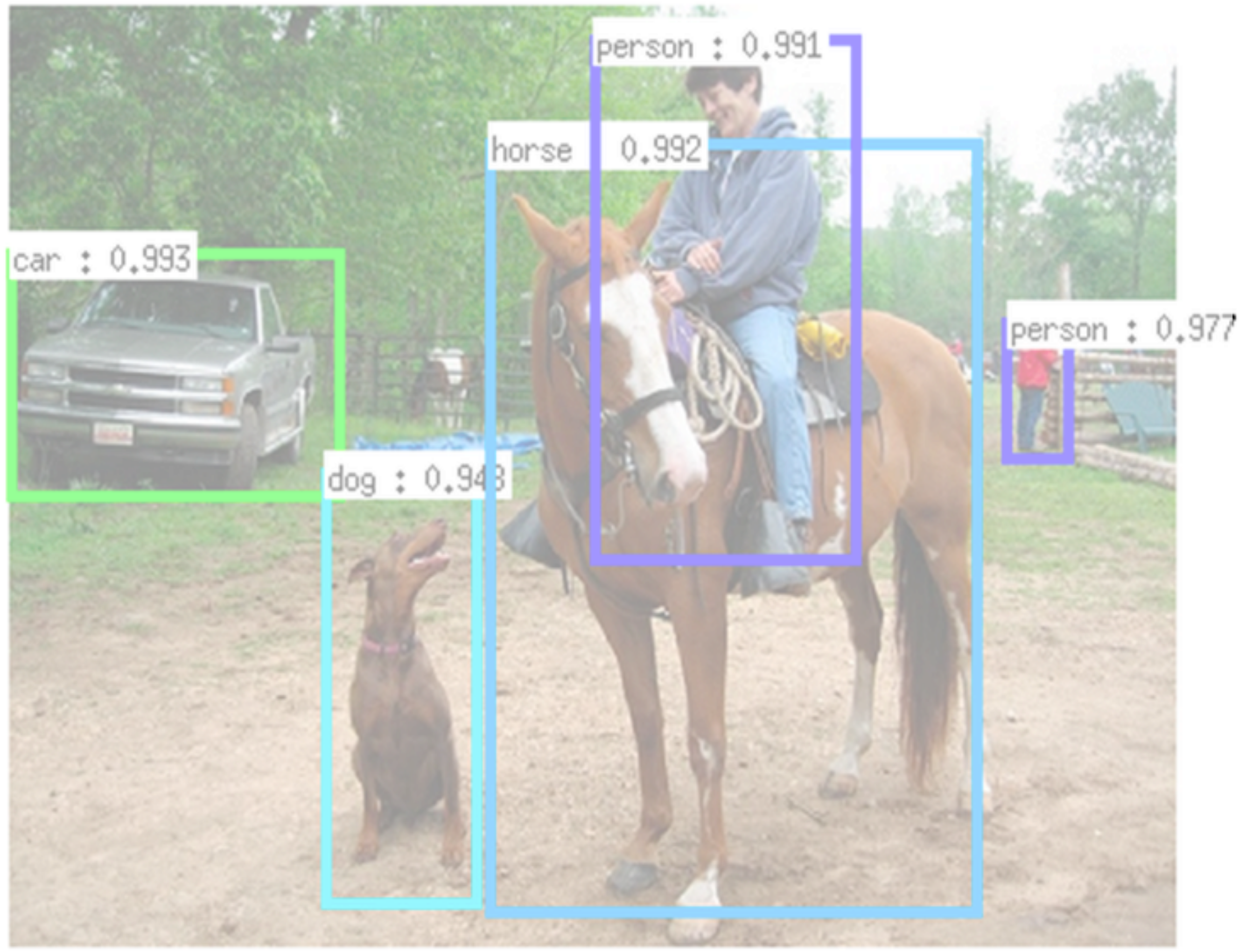
Initialization	Train Set	
	VOC07	VOC07+12
ImageNet Supervised	68.5	75.8
Places Supervised	65.3	73.1
Jigsaw ImageNet-1k	56.6	64.7
Jigsaw ImageNet-22k	67.1	73.0
Jigsaw YFCC100M	62.3	69.7

within error

Surface Normal Estimation

Image classification
Few-shot learning

ImageNet, Places-205, VOC’07, COCO



Object detection
VOC’07

3D Understanding
Surface Normals – NYUv2

Navigation
Gibson environment

Surface Normal Estimation

- Predict surface normals on NYU-v2
  - Same optimization parameters for all methods (including supervised)
  - PSPNet Architecture
  - Train last few layers only (res5 onwards)

Input

Output

Image from the NYU dataset

Surface Normal Estimation

Initialization	Median Error	% correct within 11.25^0
ImageNet Supervised	17.1	36.1
Places Supervised	14.2	41.8
Jigsaw ImageNet-1k	14.5	41.2
Jigsaw ImageNet-22k	13.4	43.7
Jigsaw YFCC100M	**13.1**	**44.6**

+8

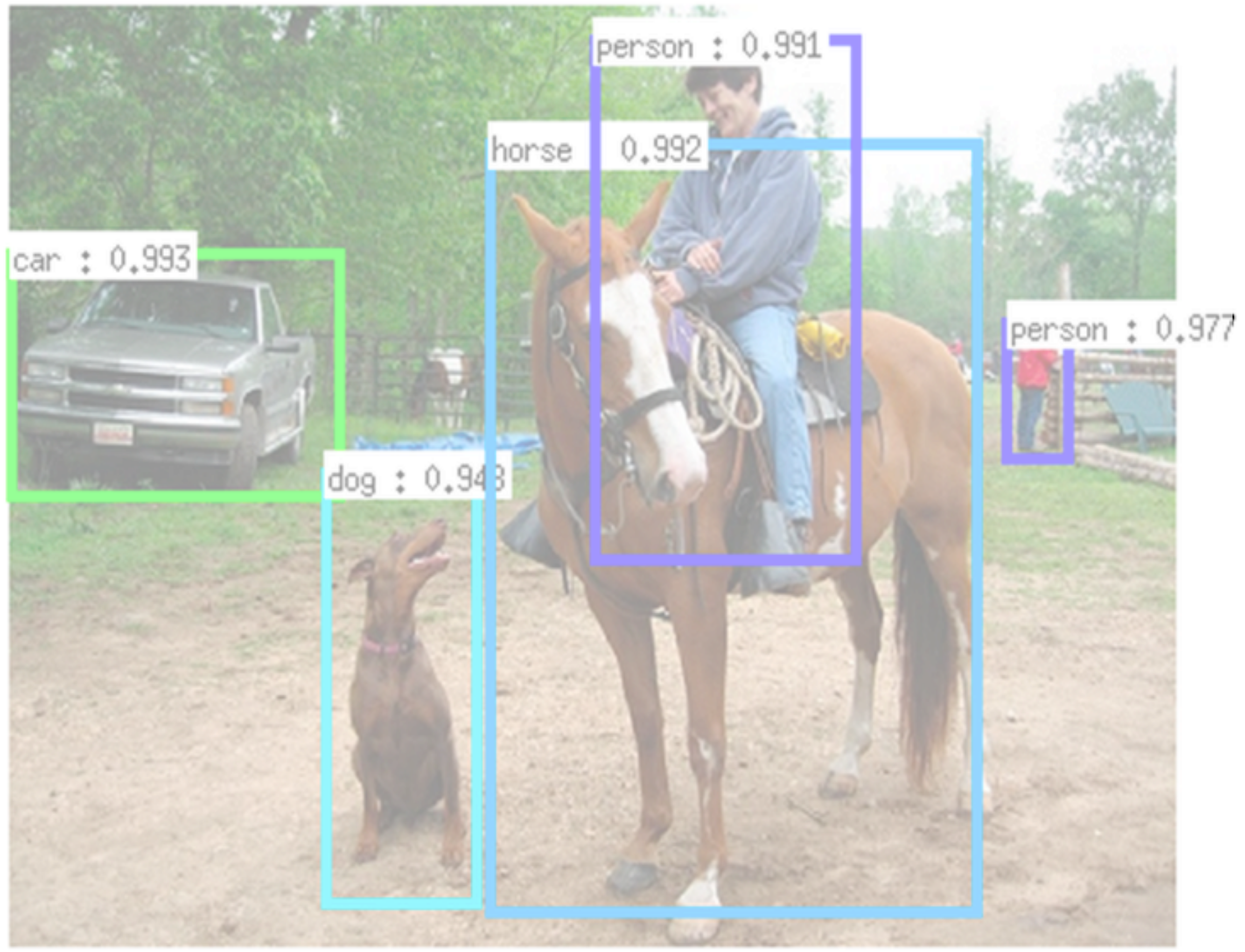
facebook
Artificial Intelligence Research

Surface Normal Estimation

Image classification

Few-shot learning

ImageNet, Places-205, VOC'07, COCO



Object detection

VOC'07

3D Understanding

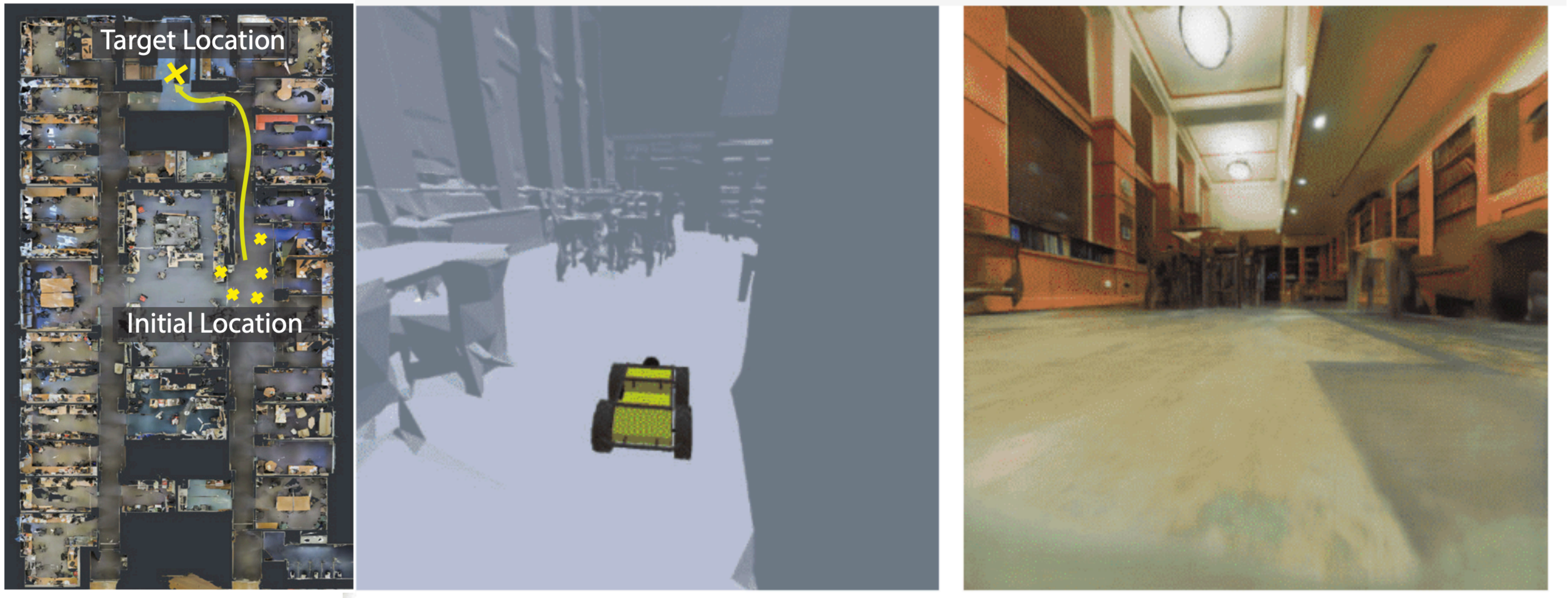
Surface Normals – NYUv2

Navigation

Gibson environment

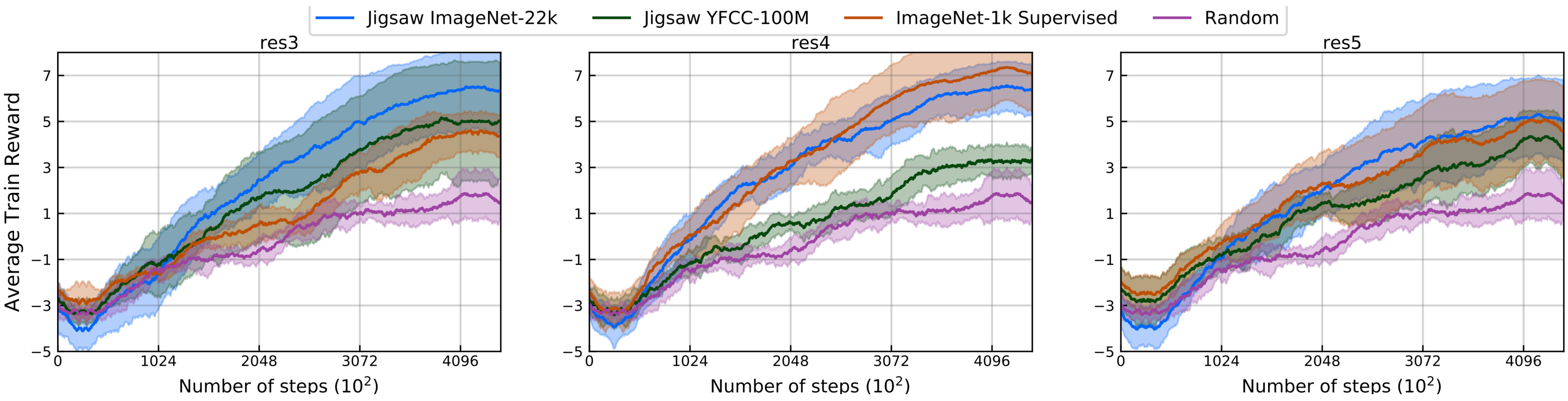
Visual Navigation

- Visual navigation in the Gibson environment
  - Method from Sax et al., 2018
  - Fixed ConvNet features

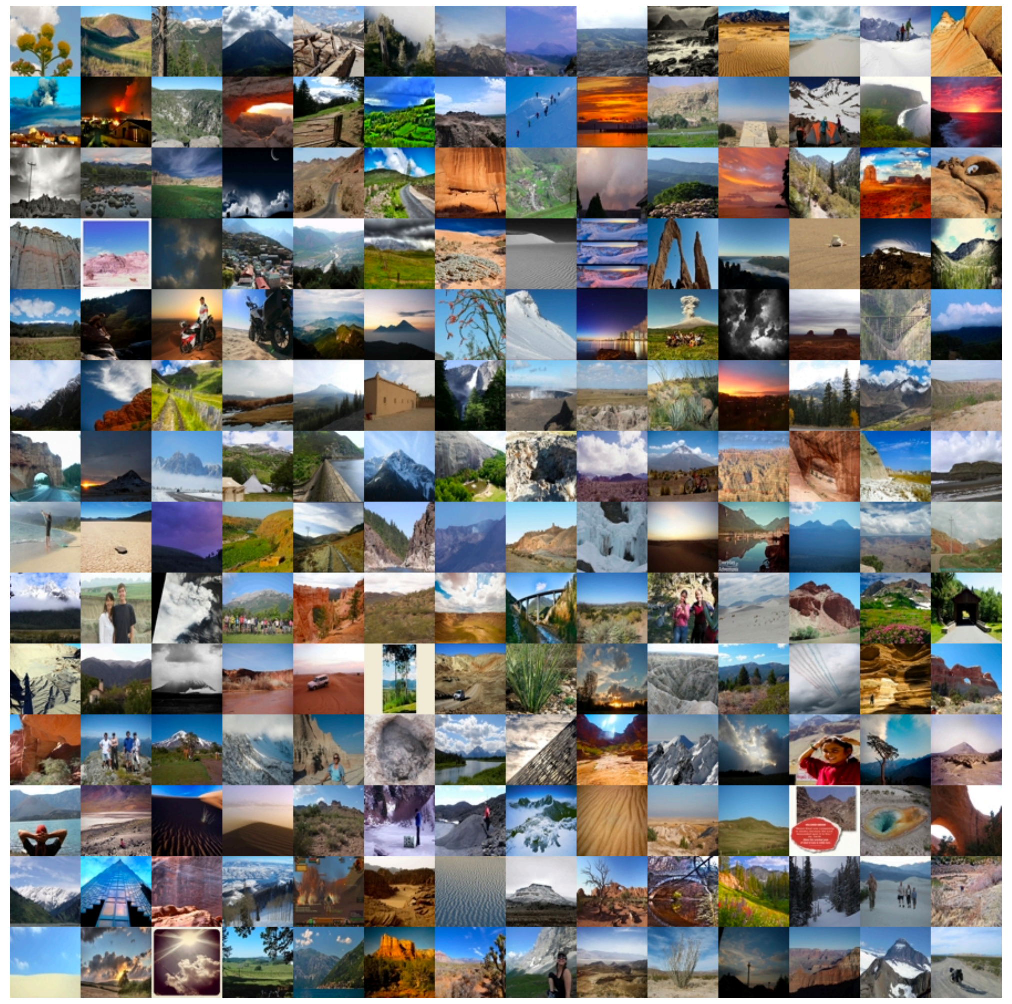


Gibson Environment

Visual Navigation

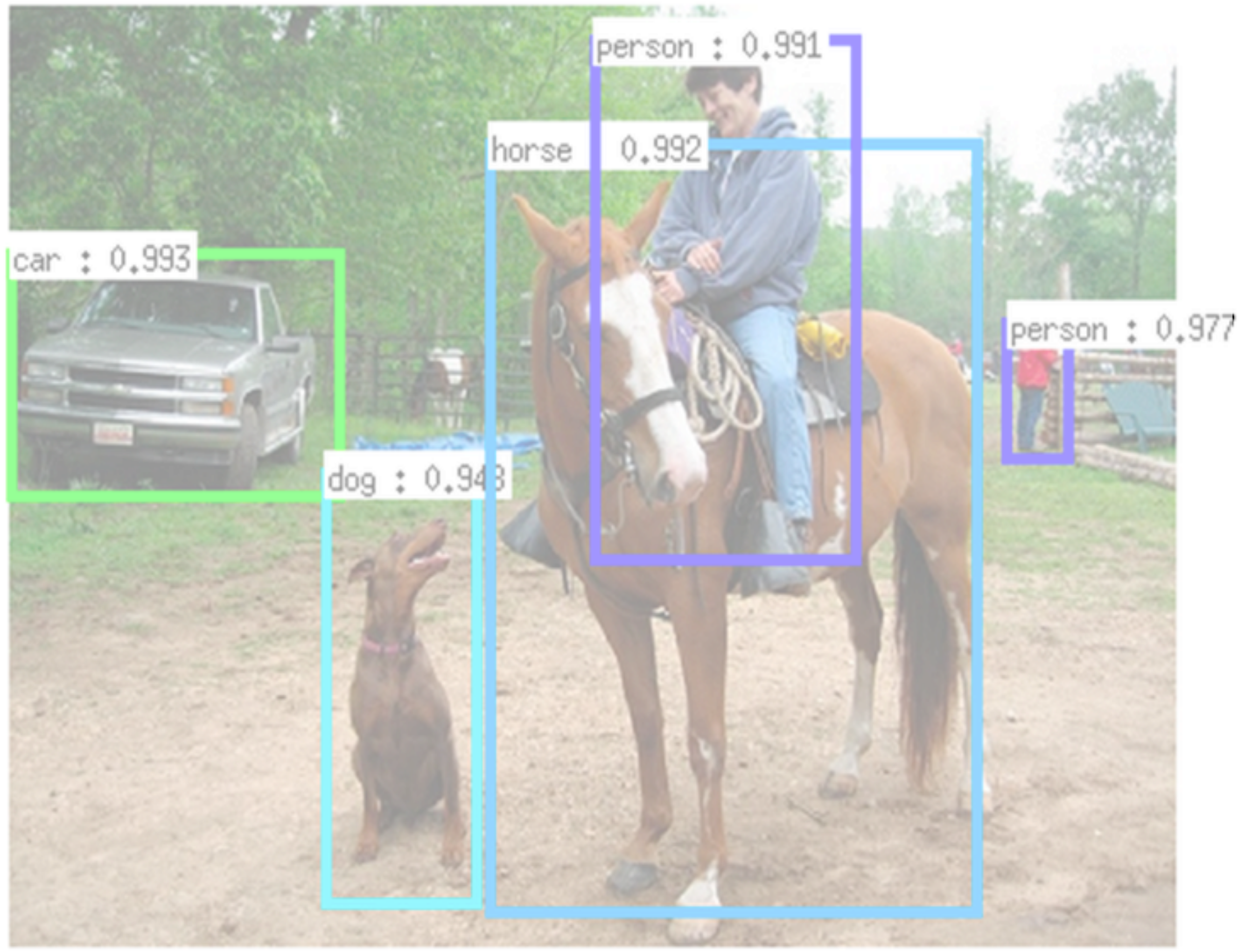


Few shot learning



Few-shot learning

Places-205, VOC’07



Object detection

VOC’07

3D Understanding

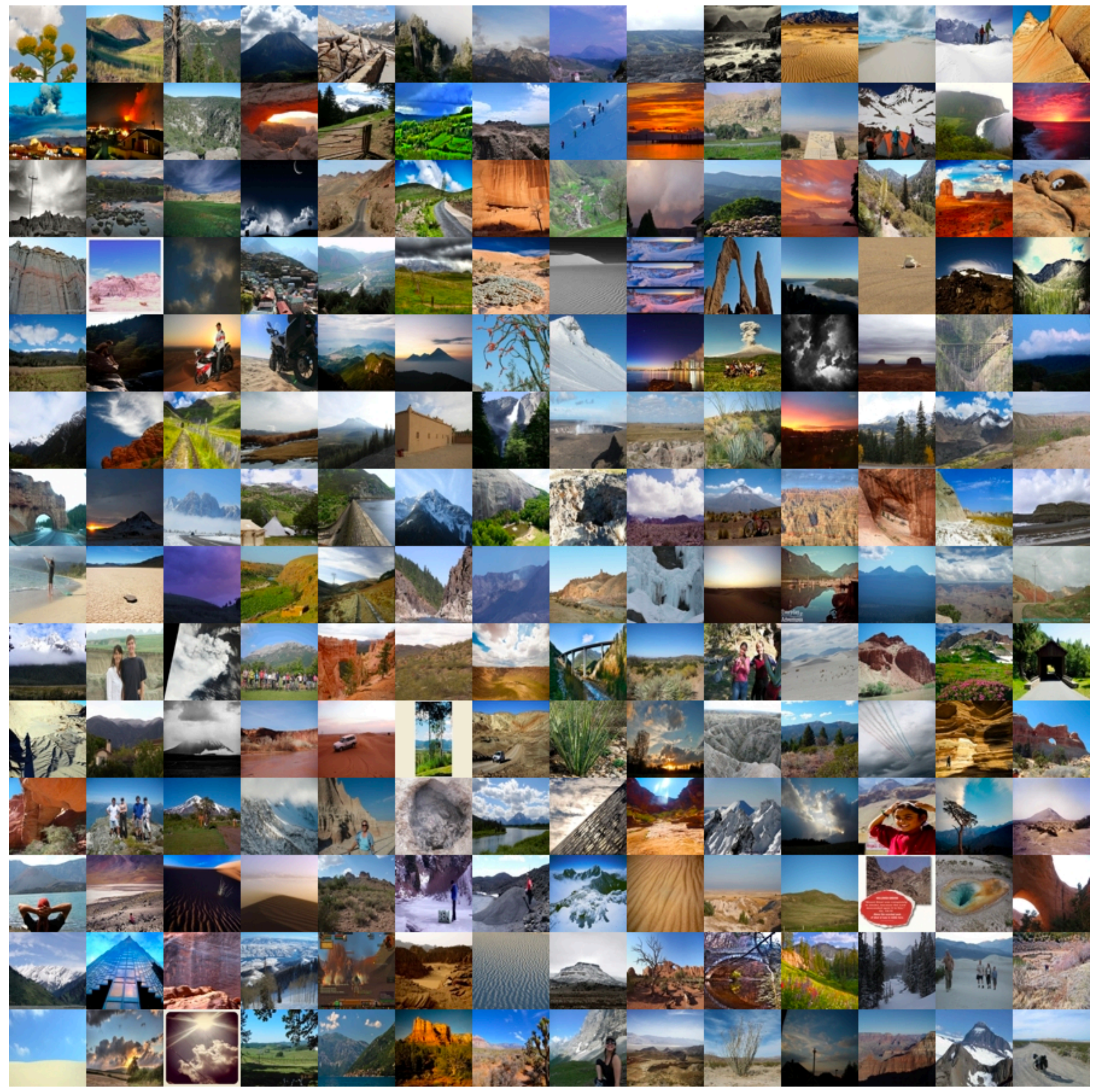
Surface Normals – NYUv2

Navigation

Gibson environment

Few shot learning

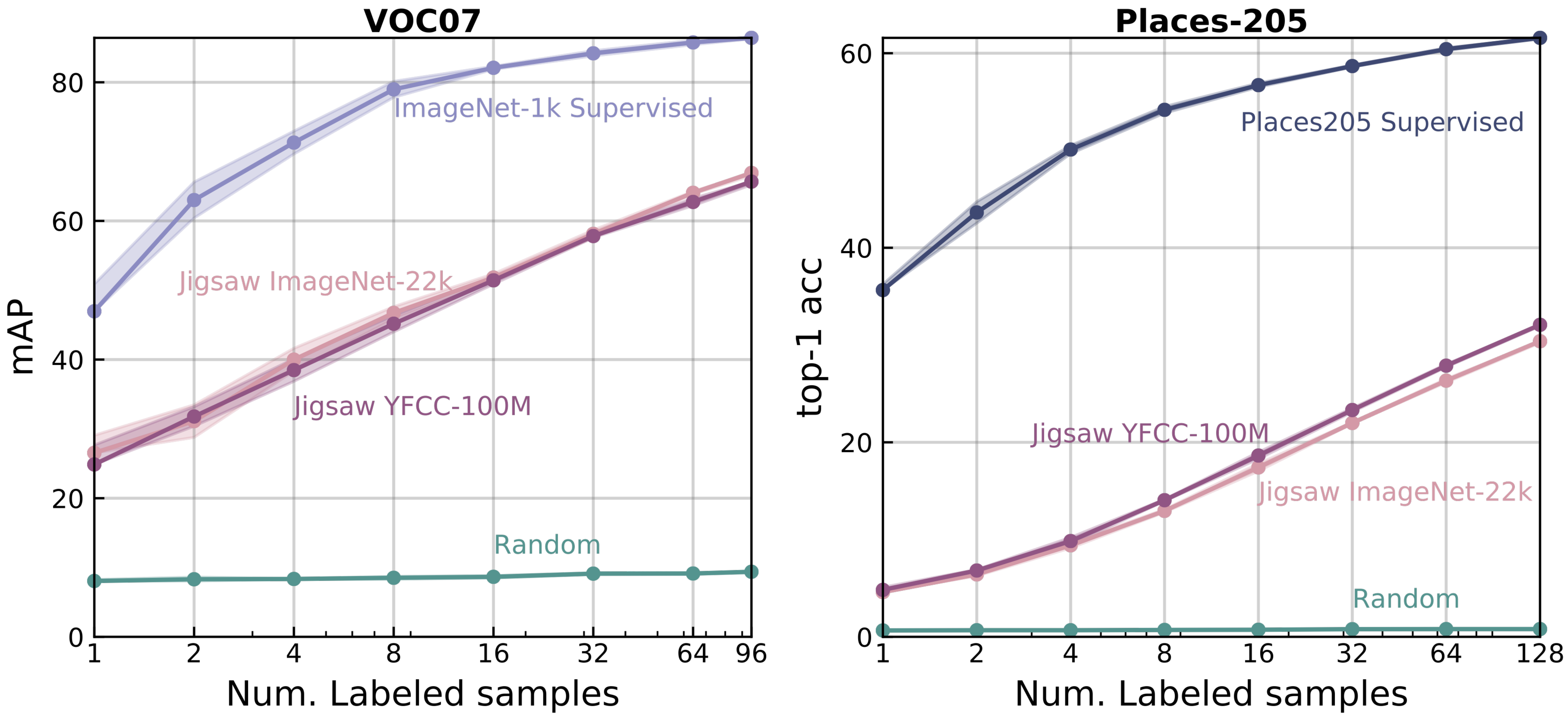
- k-shot learning
  - Use VOC’07/Places205 classification
  - K labeled examples per class
  - Train linear SVMs



Few-shot learning

Image from the Places dataset

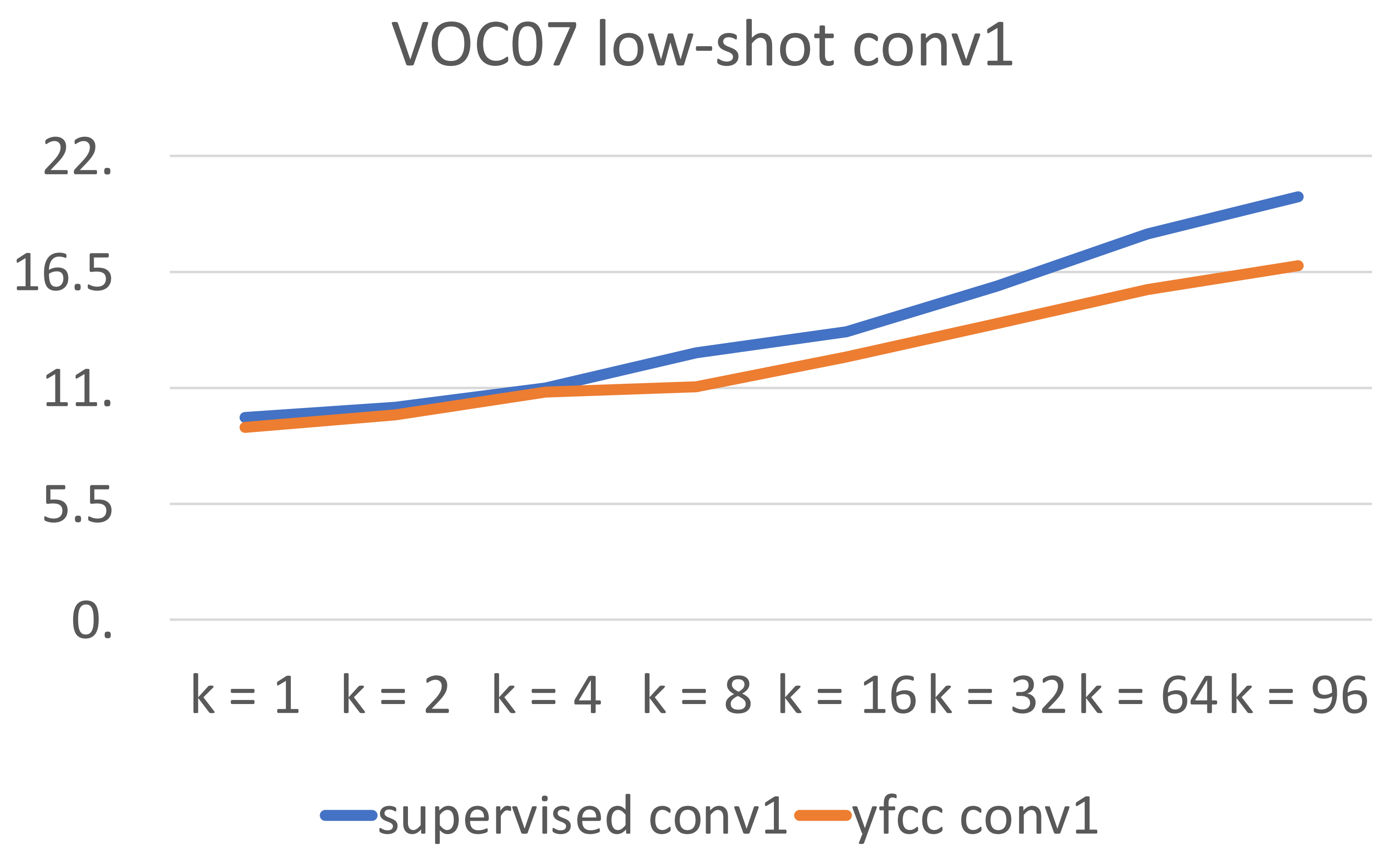
Few shot learning

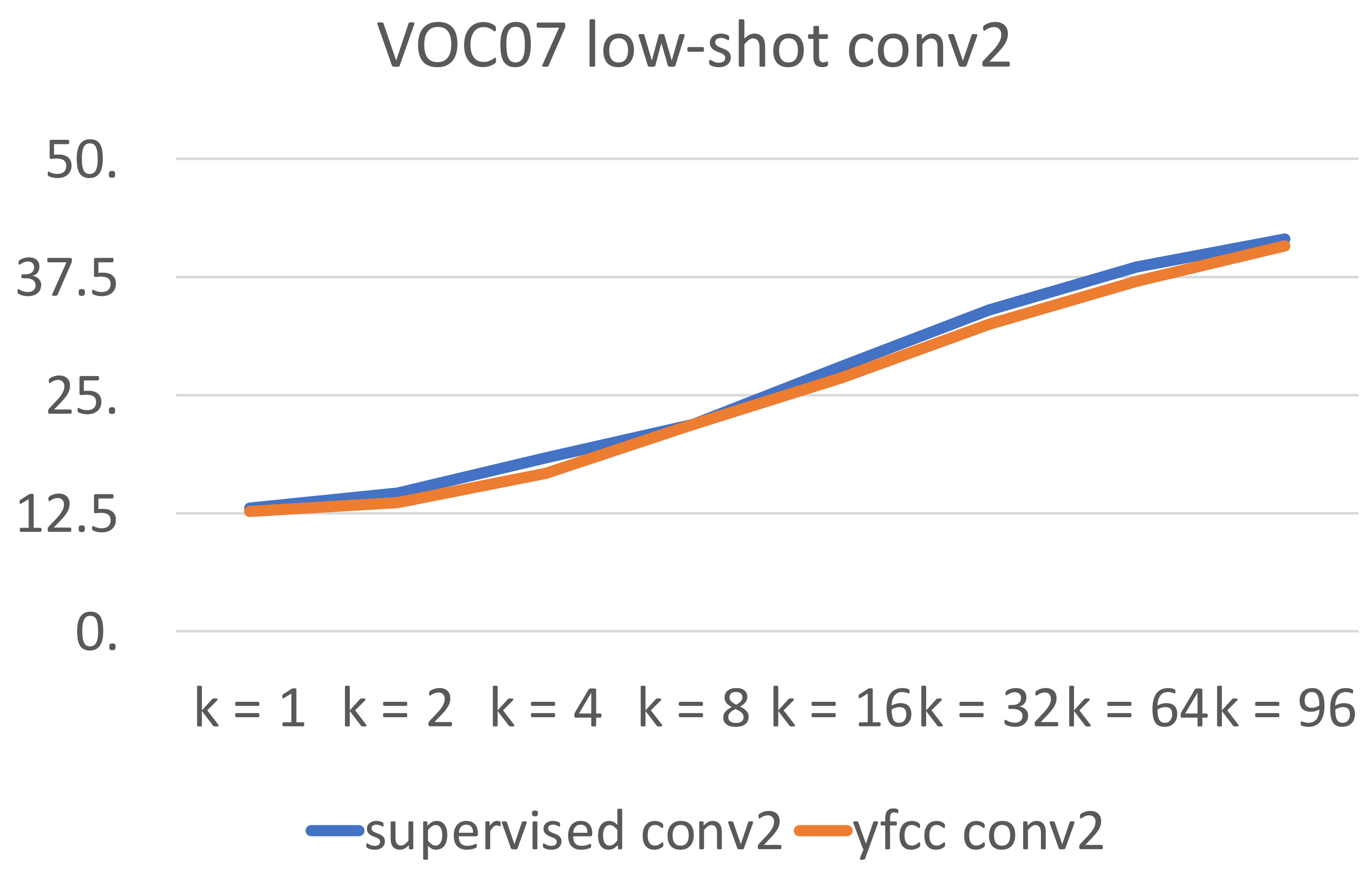


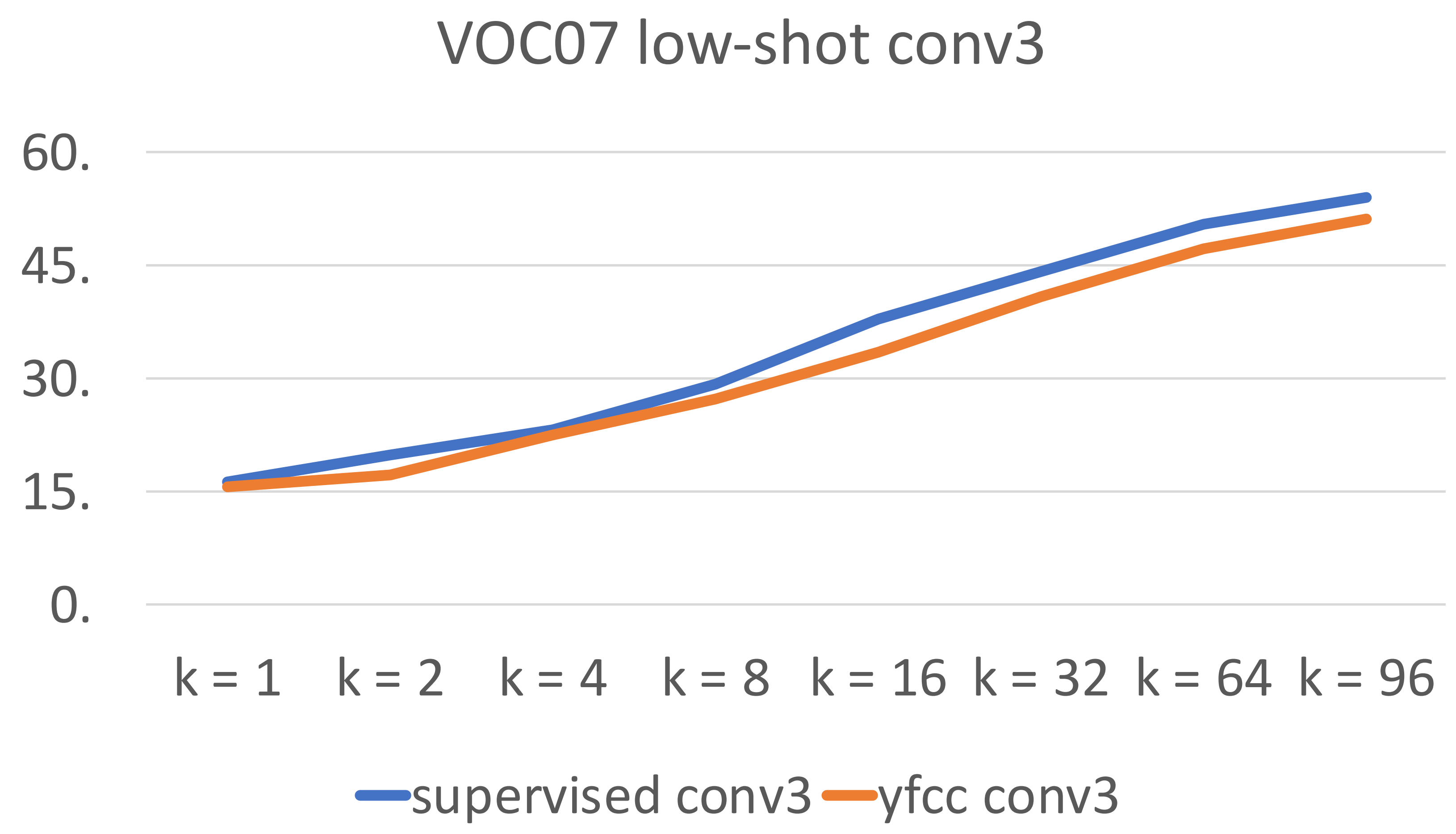
facebook
Artificial Intelligence Research

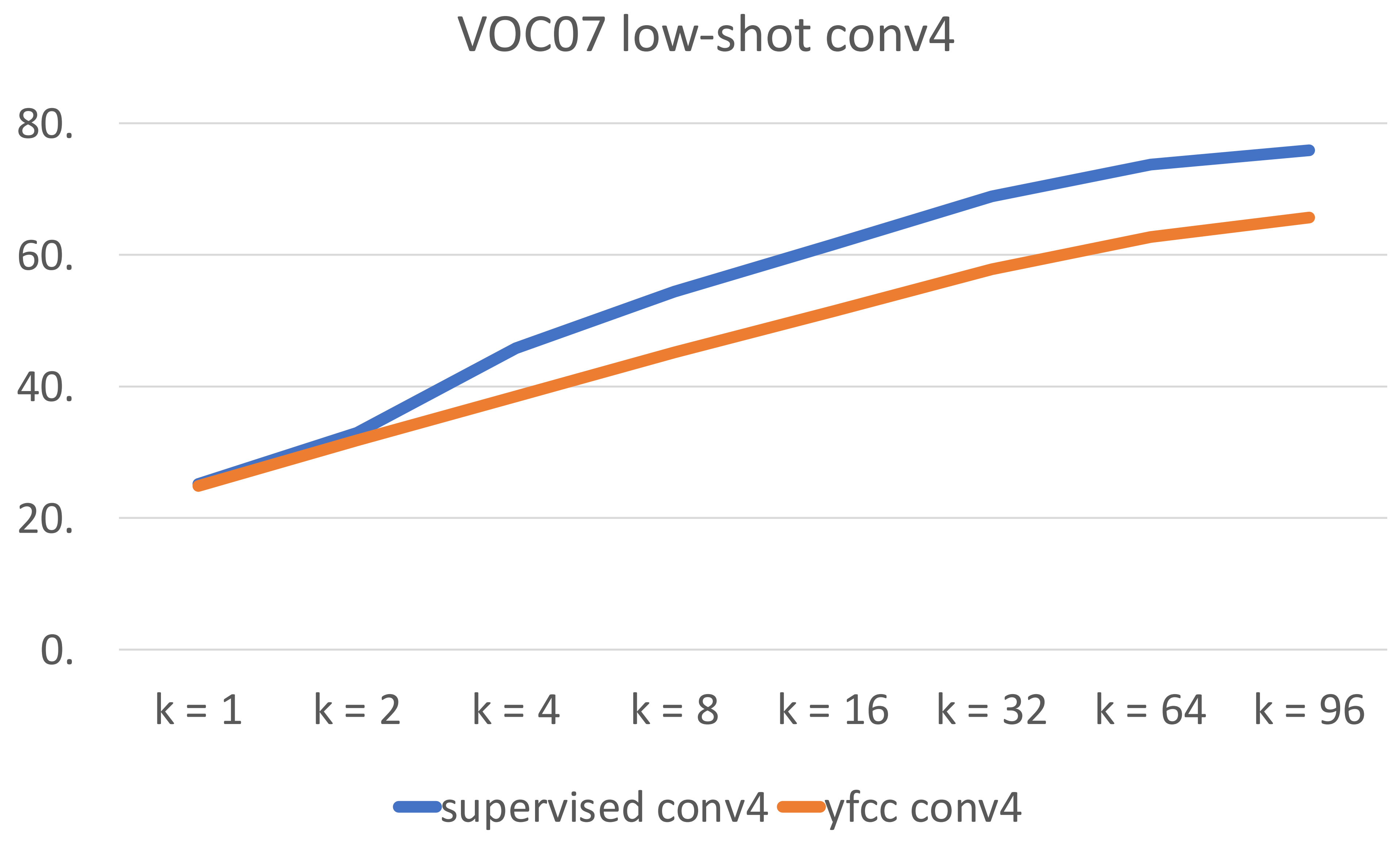
Self-supervised representations are not as sample efficient

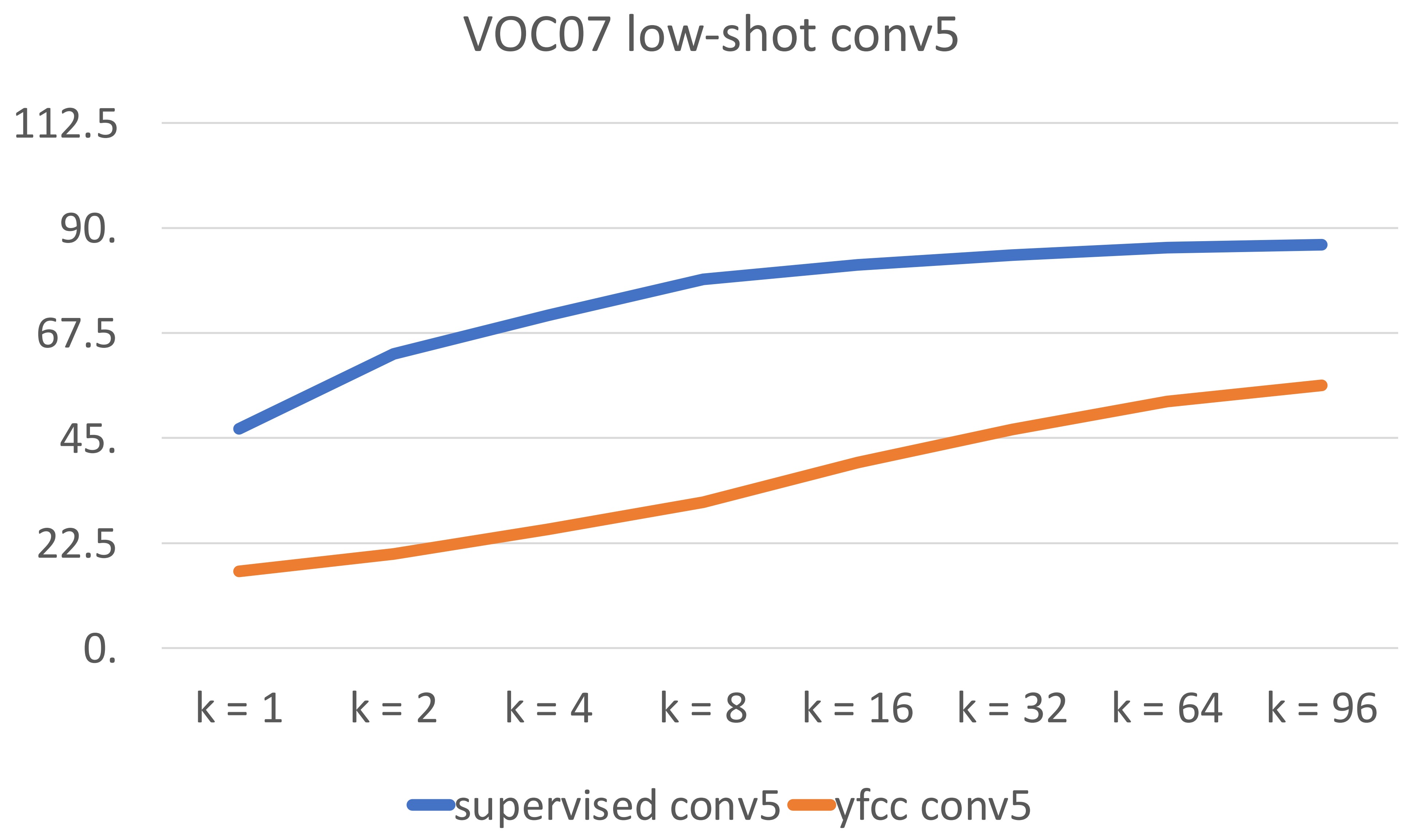
Few shot learning - VOC07



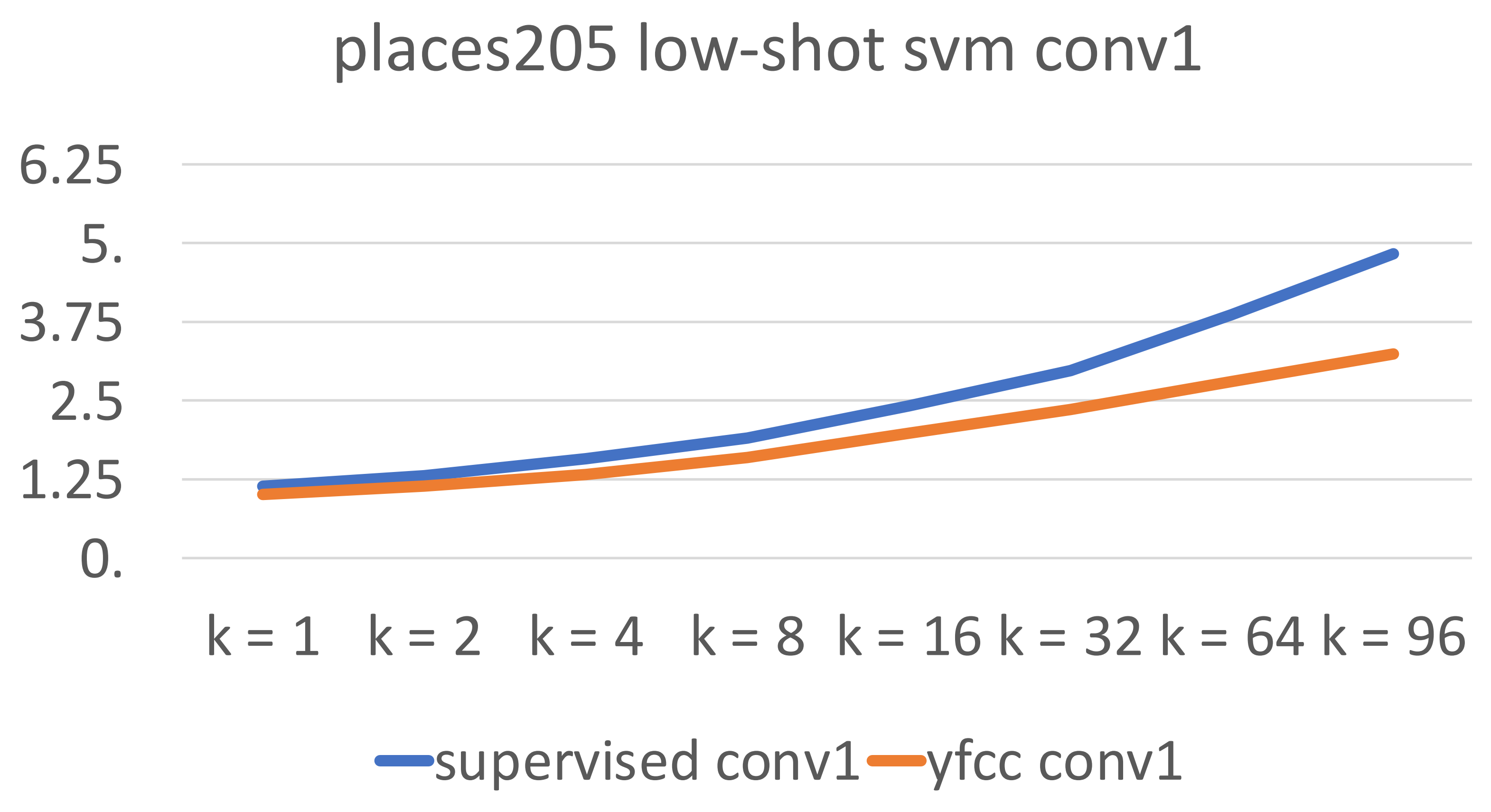


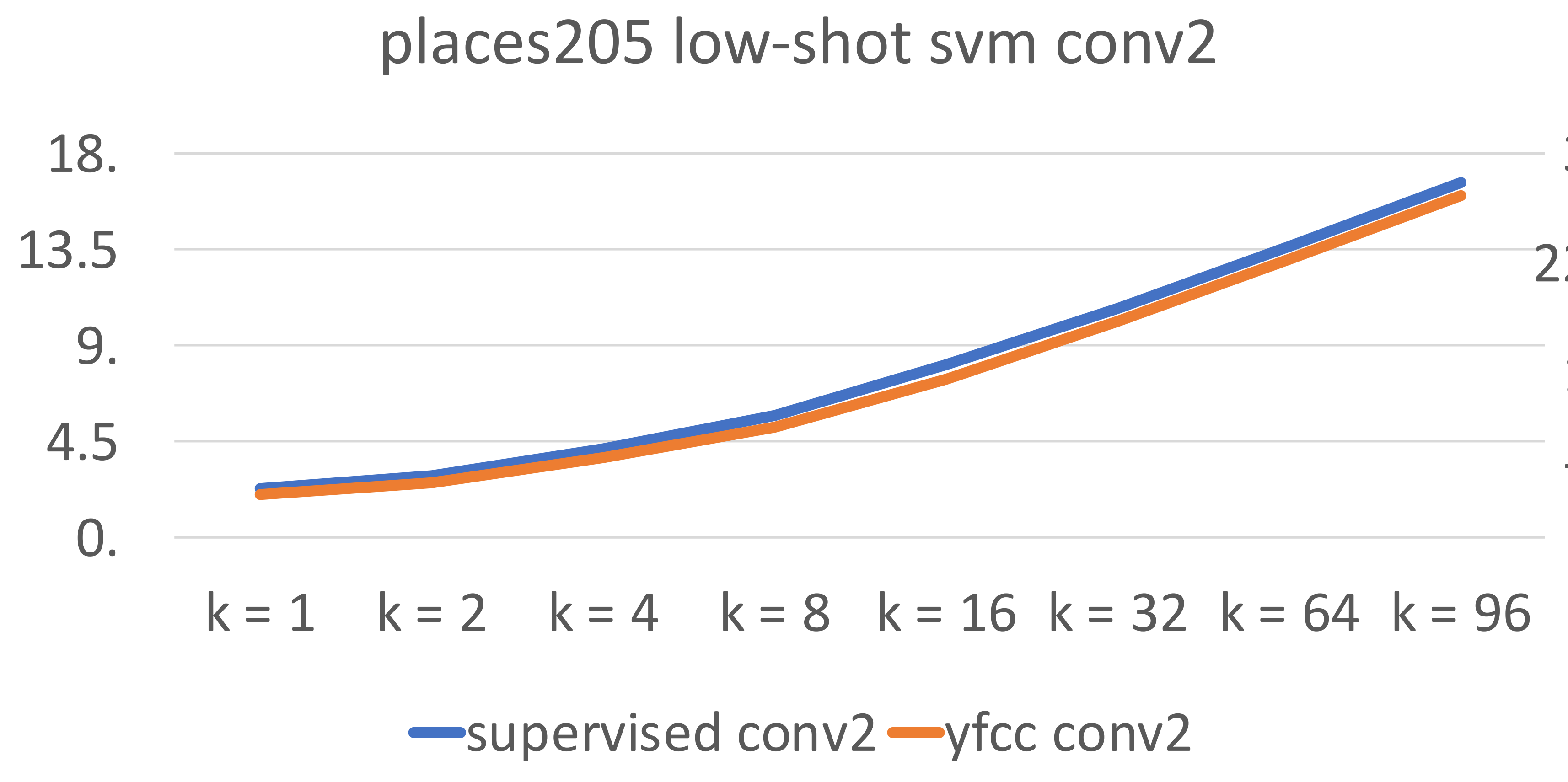


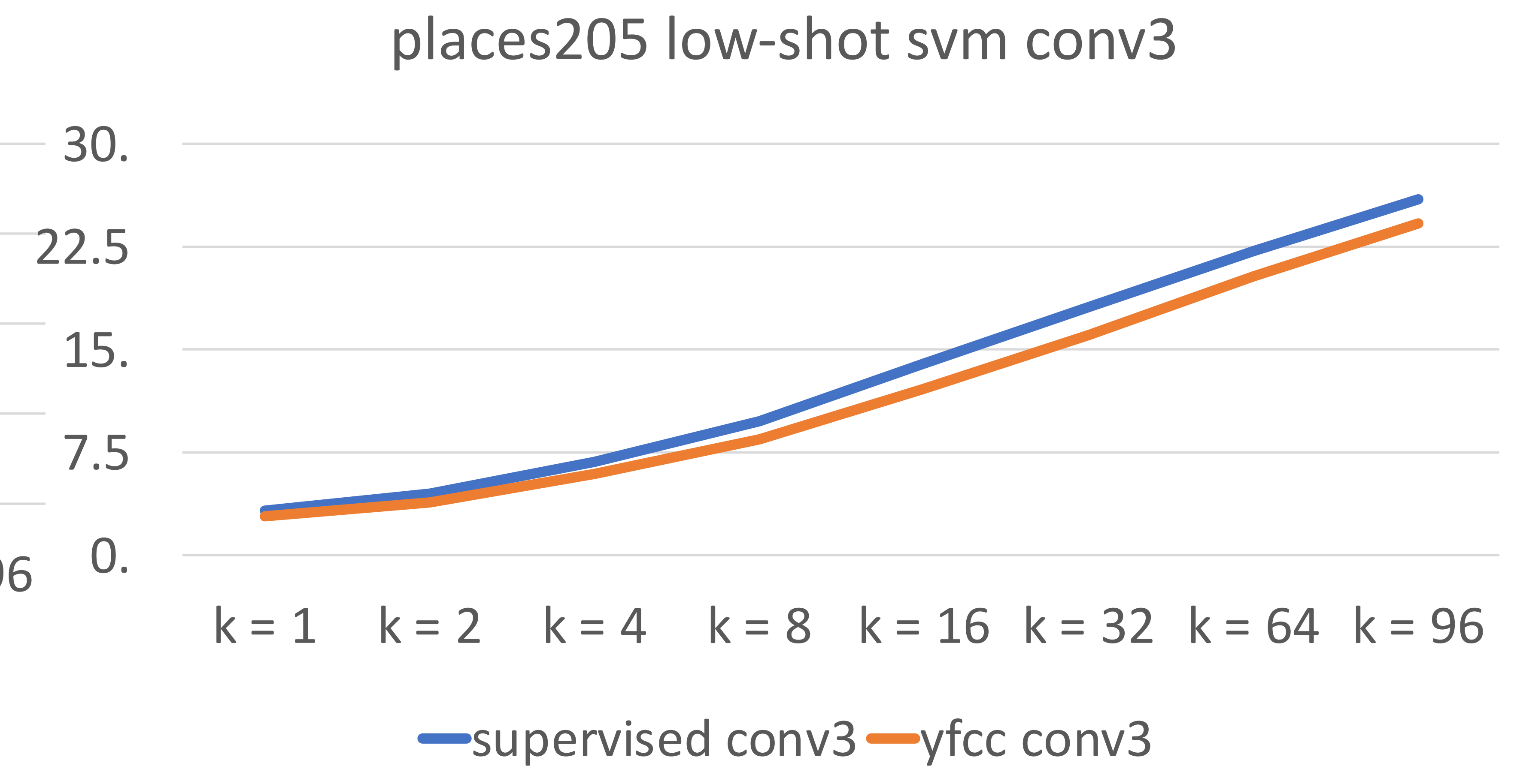


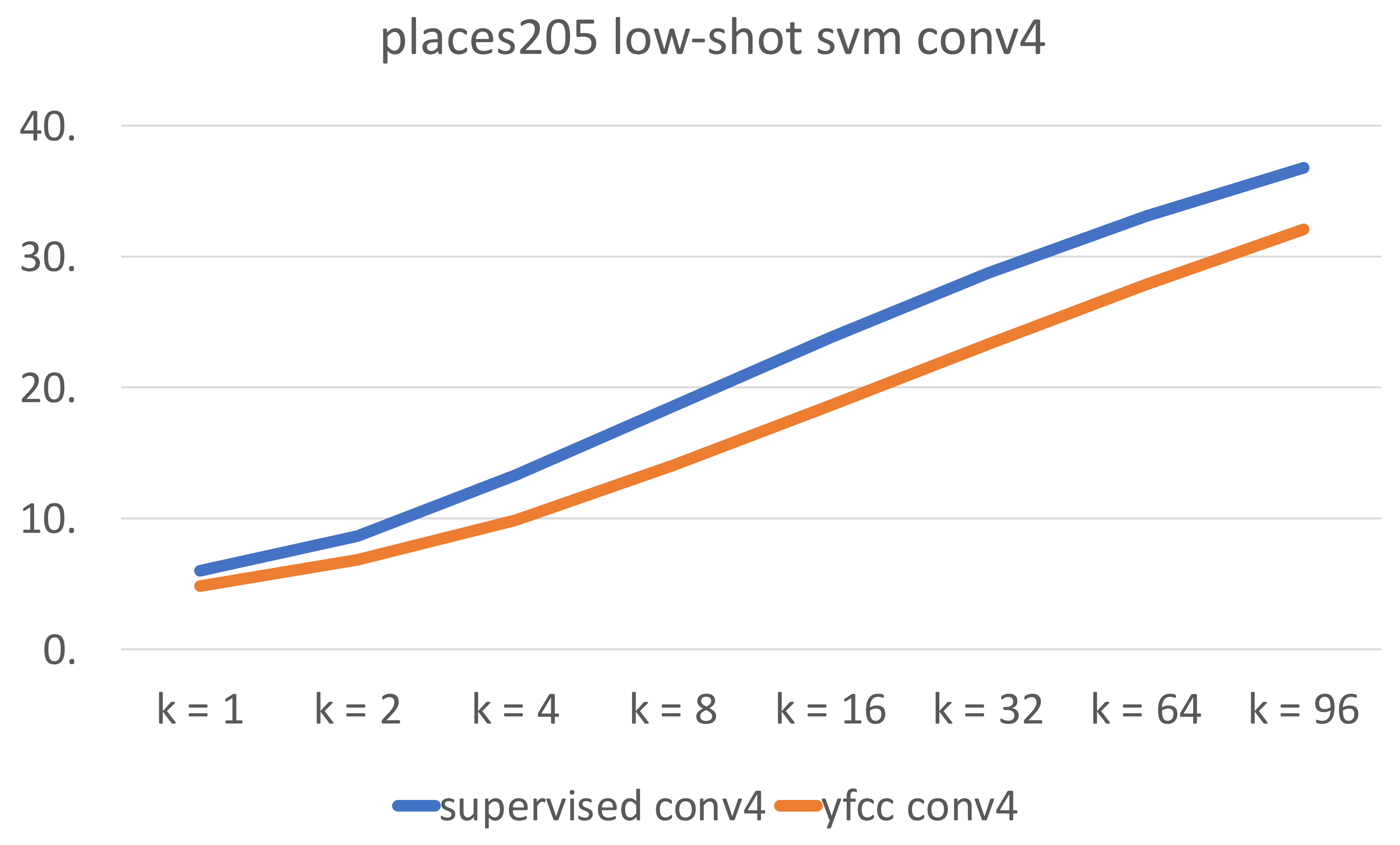


Few shot learning - Places205









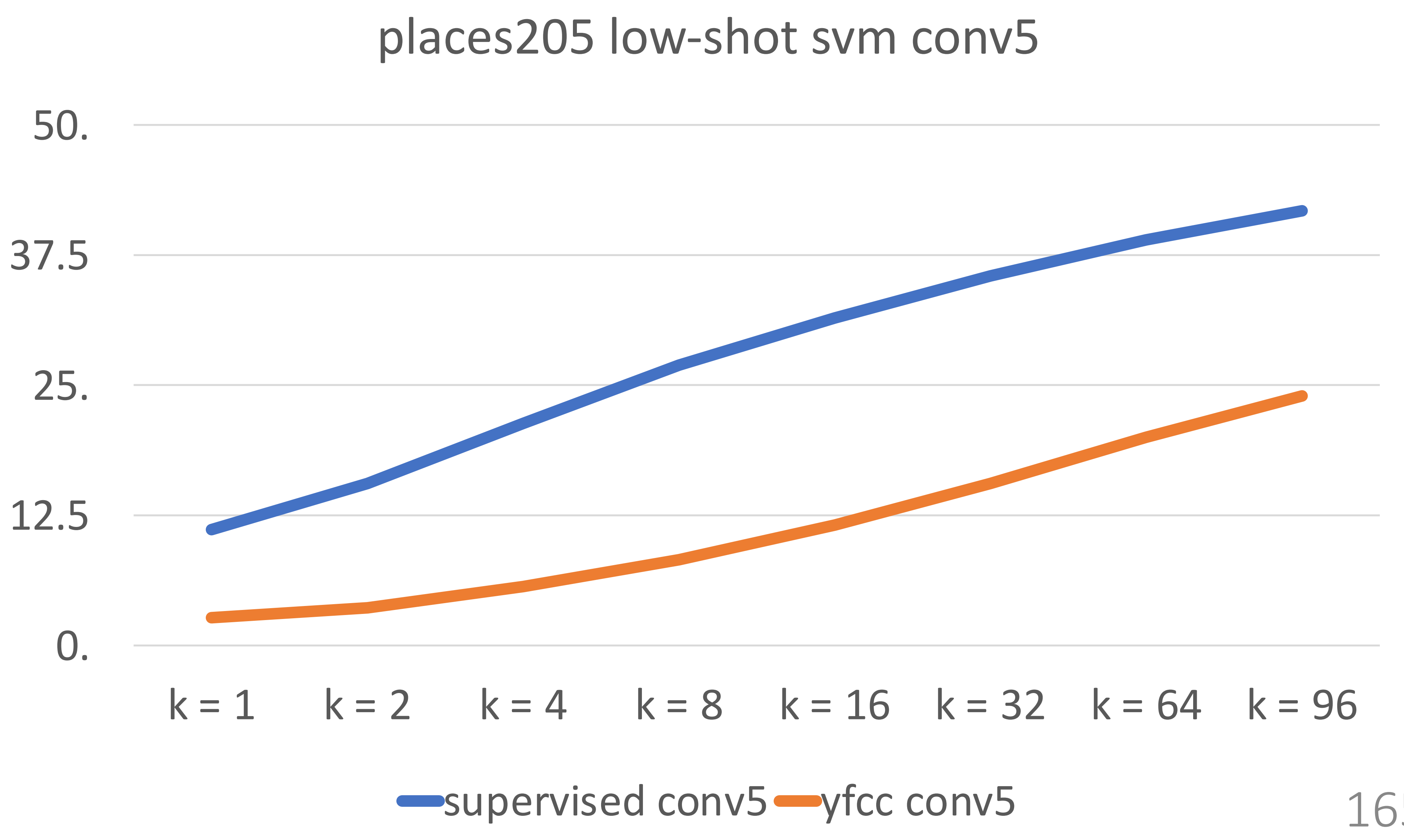


Image Classification

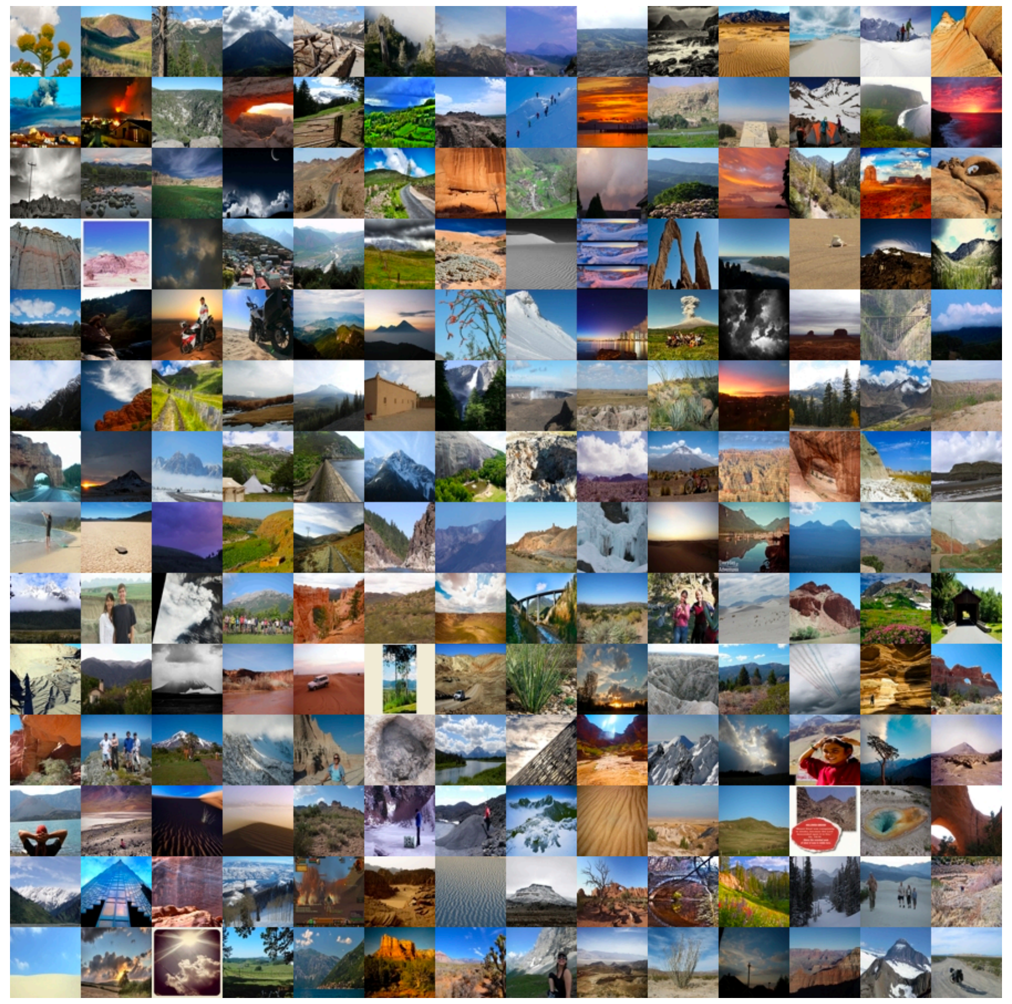
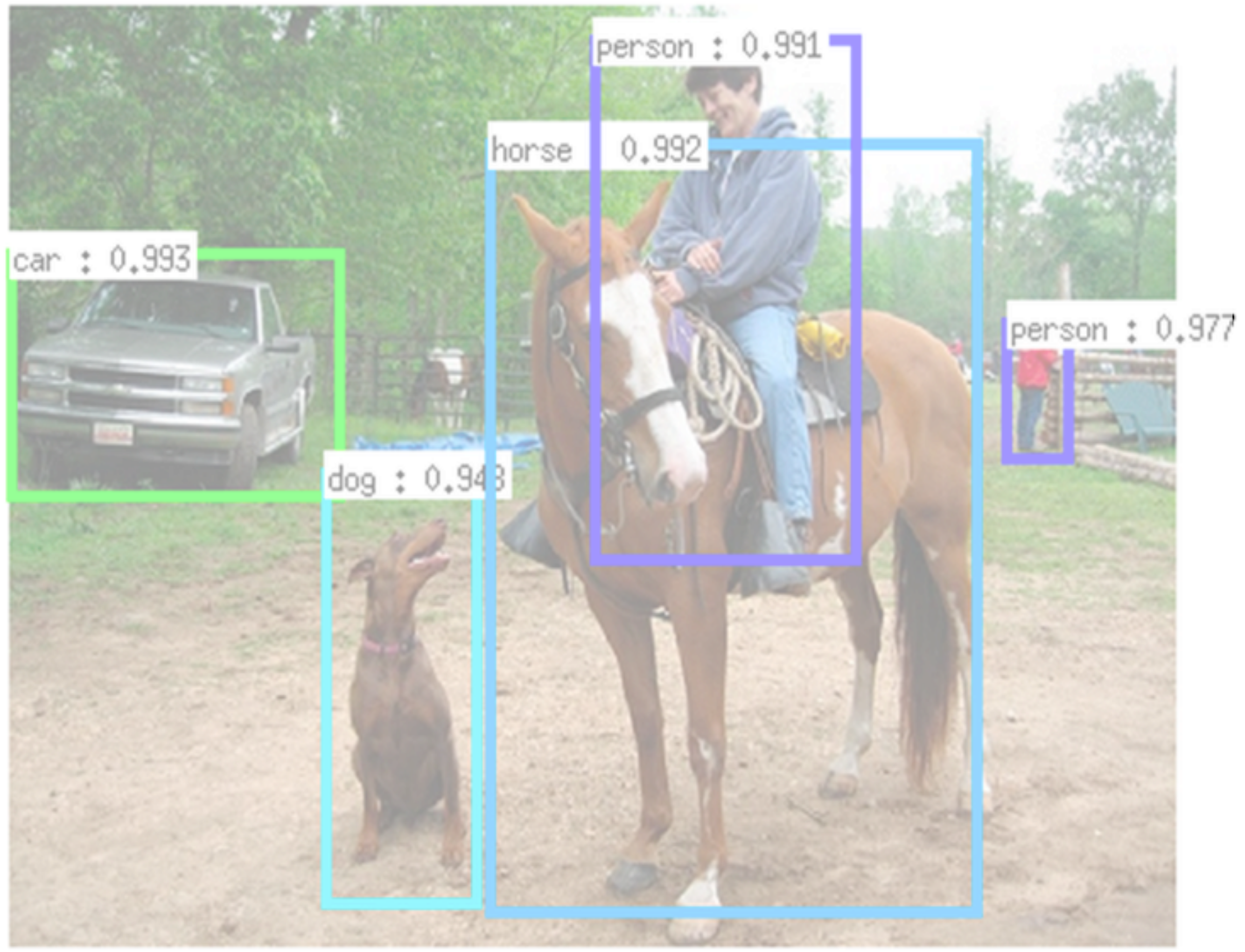


Image classification

Places-205, VOC'07



Object detection

VOC'07

3D Understanding

Surface Normals – NYUv2

Navigation

Gibson environment

Linear SVMs on VOC07

VOC2007 SVM classification. ResNet50

Init	conv1	stage1	stage2	stage3	stage4
ImageNet supervised	24.49	47.75	60.54	80.36	87.95
ImageNet jigsaw	27.09	45.73	56.61	64.51	57.17
Imagenet14M jigsaw	23.46	46.72	58.52	71.76	64.92
YFCC100M jigsaw	18.98	46.71	57.77	71.21	64.34

- Deeper self supervised layers are less transferable
- Hypothesis – Problem is not "complex" enough.

SGD-based Linear Classifiers on Places205

Places205 linear classification. ResNet50

Init	conv1	stage1	stage2	stage3	stage4
ImageNet supervised	14.84	32.59	42.06	50.83	**52.49**
ImageNet jigsaw	15.079	28.753	36.825	**41.232**	34.364
Imagenet14M jigsaw	13.973	29.462	36.656	**41.721**	36.254
YFCC100M jigsaw	10.252	29.672	39.565	**44.866**	38.219

- Gap between ImageNet and self-supervised methods is smaller.
- Places has scenes while ImageNet is object centric
- YFCC is a good mixture of both.

Evaluation: Main Lessons

- Evaluation on multiple tasks is essential
- Evaluation with fixed features or at least same hyper-parameters
- Evaluate sample efficiency of representations

What's missing from self-supervised methods?

- Complex problems, big data and deeper models
- Current self-supervised methods do not seem to learn high level representations
- Sample efficiency

Thanks!

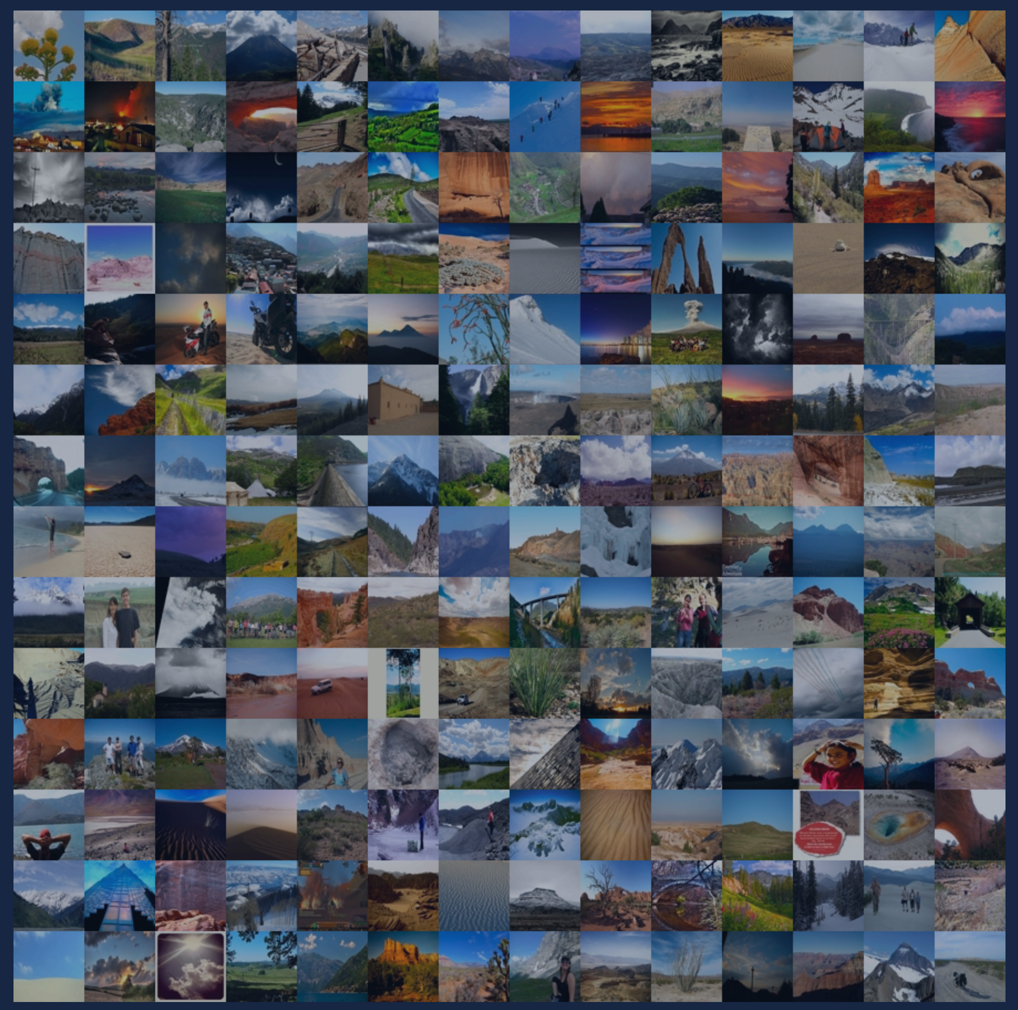
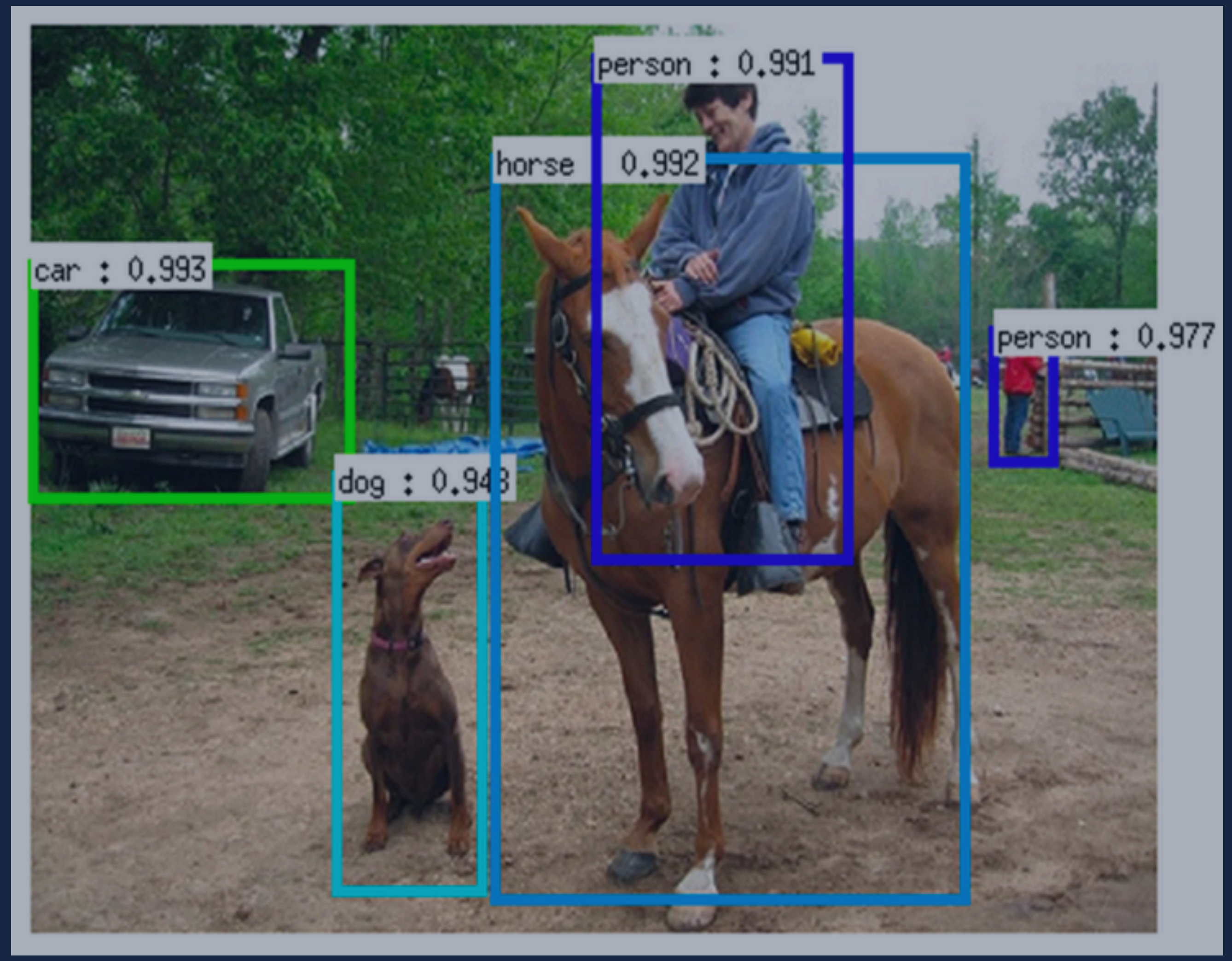


Image classification

Few-shot learning

ImageNet, Places-205, VOC'07, COCO



Object detection

VOC'07

3D Understanding

Surface Normals – NYUv2

Navigation

Gibson environment

facebook

Artificial Intelligence Research